



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA

**Identificació i control d'una planta dissenyada  
per validar el control repetitiu**

**Enric Xargay Mata, Ramon Costa Castelló**

*IOC-DT-P-2005-4  
Gener 2005*



# **Identificació i Control d'una planta dissenyada pel validar el control repetitiu.**

**Enric Xargay Mata & Ramon Costa Castelló**

Novembre 2004

## **ÍNDEX**

<b>Capítol 1: Introducció .....</b>	<b>2</b>
1.1. Objectiu .....	2
1.2. Organització de la memòria .....	2
<b>Capítol 2: Descripció de la planta.....</b>	<b>3</b>
<b>Capítol 3: Identificació de la planta .....</b>	<b>5</b>
3.1. Introducció .....	5
3.2. Identificació de sistemes dinàmics.....	5
3.3. Corba estàtica .....	6
3.4. Resposta temporal .....	7
3.5. Model lineal: Dynamic Signal Analyzer .....	9
3.6. Model lineal: System Identification Toolbox (MatLab) .....	11
3.7. Model lineal: equacions físiques .....	19
3.8. Model complet.....	21
<b>Capítol 4: Control repetitiu .....</b>	<b>26</b>
4.1. Introducció .....	26
4.2. Model de la pertorbació .....	27
4.3. Sistema de control .....	29
4.4. Disseny del controlador .....	30
4.5. Simulacions .....	30
<b>Conclusions .....</b>	<b>42</b>
<b>Bibliografia .....</b>	<b>43</b>
<b>Annex A: Codis MatLab .....</b>	<b>44</b>
<b>Annex B: Índex CD.....</b>	<b>58</b>

## **Capítol 1**

### **INTRODUCCIÓ**

#### **1.1. Objectiu**

L'objectiu principal d'aquest treball és l'implementació d'un sistema de control basat en la Teoria de Control Repetitiu que permeti resoldre el problema del rebuig de perturbacions periòdiques de freqüència constant.

El projecte comprèn també la determinació del model d'una planta d'experimentació ja existent, que ha de servir com a plataforma per a la validació dels resultats teòrics obtinguts en aquest treball.

#### **1.2. Organització de la memòria**

Aquesta memòria està estructurada en tres grans blocs. En el primer bloc es descriu el sistema a controlar. En el segon bloc s'obtenen tres models de la planta mitjançant tres metodologies diferents: amb l'ajuda d'un *dynamic signal analyzer*, mitjançant la *toolbox* d'identificació de sistemes de MatLab i a partir de les equacions físiques. En el tercer bloc s'introdueix de forma molt breu la teoria del control repetitiu, es descriu un esquema per resoldre el rebuig de perturbacions de freqüència constant i coneguda, i es realitzen diverses simulacions per a comprovar la validesa d'aquest tipus de control.

## Capítol 2

### DESCRIPCIÓ DE LA PLANTA

La planta està formada per un motor de corrent continu (PMDC<sup>1</sup>) al qual s'ha acoblat un dispositiu magnètic, que provoca un parell variable sobre l'eix del motor que actua com a pertorbació (figures 1 i 2). Es pretén controlar la velocitat de gir de l'eix del motor.

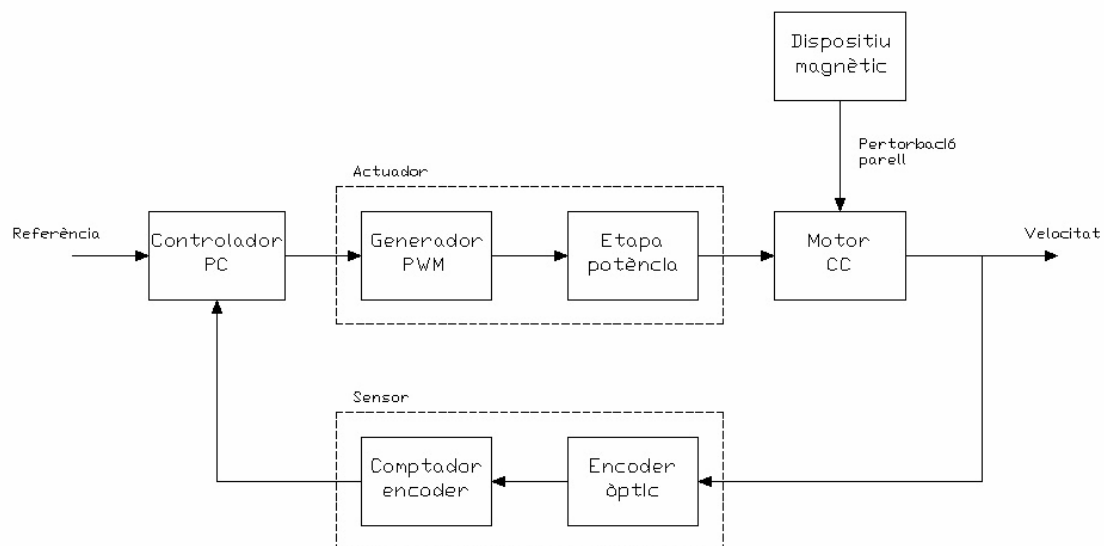


Figura 1: Esquema de blocs del sistema

Concretament, el dispositiu magnètic està constituït per una parella d'ímants solidaris a l'eix del motor que interaccionen amb una altra parella d'ímants fixos, tots situats al mateix pla (figura 2).

Així, la rotació de l'eix del motor provoca, a causa de l'interacció entre els ímants mòbils i fixos, un parell pulsant sobre el propi eix (figura 3). Per a una consigna constant (graó), aquest parell pulsant és de naturalesa periòdica i la seva freqüència fonamental coincideix amb la velocitat de rotació de l'eix del motor.

<sup>1</sup> De l'anglès, *Permanent Magnet Direct Current*.



Figura 2: Fotografia de la planta

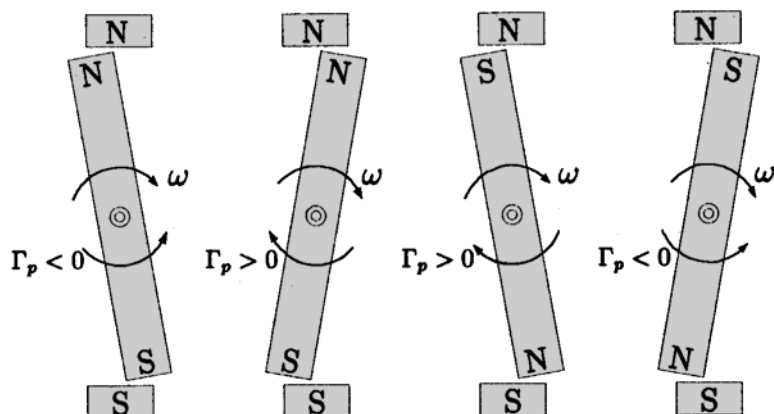


Figura 3: Esquema del dispositiu magnètic que provoca el parell de perturbació sobre l'eix del motor

El motor està governat des d'un PC que fa de controlador, a través d'un generador PWM i una etapa de potència.

Per la detecció de la velocitat de gir, es disposa d'un encoder òptic de dos canals i 500 polsos per volta i canal acoblat a l'eix del motor. La resolució de l'encoder es multiplica per quatre gràcies a un decodificador de quadratura, que, a més, permet conèixer el sentit de gir del motor. El senyal del decodificador s'introdueix a un comptador que, mostrejat periòdicament, permet determinar la velocitat de gir del motor a nivell software.

### **Capítol3**

## **IDENTIFICACIÓ DE LA PLANTA**

### **3.1. Introducció**

En automàtica, l'obtenció d'un model fiable que descriu de forma correcta la dinàmica del sistema a controlar és de vital importància. És usual realitzar simulacions de l'algorisme de control contra un model matemàtic de la planta amb l'ajuda de paquets informàtics tipus MatLab de forma prèvia a la implementació del controlador. Procedint d'aquesta manera, es redueix el risc de fer malbé el sistema de control.

En el cas del control repetitiu, a més, el disseny de l'algorisme de control requereix el coneixement d'un model lineal de la planta. La precisió amb què es determini aquest model serà fonamental pel bon funcionament del controlador. Això fa que s'hagi optat per deduir el model de la planta a partir de tres procediments diferents.

Un d'aquests procediments és la determinació de les equacions físiques. Aquest tipus de modelat permet entendre el comportament del sistema i determinar les causes dels efectes que es poden observar durant l'experimentació. A més, conèixer les regles que governen el comportament del sistema també permet establir diferents tècniques i estratègies de control per a modificar aquest comportament en benefici propi.

En resum, l'objectiu d'aquest apartat és l'obtenció d'un model que descriu correctament el comportament del motor, sense tenir en compte el dispositiu magnètic causant de la pertorbació.

### **3.2. Identificació de sistemes dinàmics**

En general, un sistema es pot descriure mitjançant una part lineal, que ve descrita per una funció de transferència, i una altra part no-lineal. El fet de no incloure al model aquesta darrera part pot representar una pèrdua significativa

d'informació de la dinàmica de la planta. Tanmateix, en determinades aplicacions i per a determinats rangs de treball, l'aproximació d'un model lineal ja és suficient.

### **3.3. Corba estàtica**

La corba estàtica d'una planta és una primera aproximació al seu comportament. Concretament, presenta la relació entre les entrades i les sortides del sistema en règim permanent i dona informació sobre l'existència de no-linealitats. En el nostre cas, la corba estàtica representa la velocitat de l'eix del motor (en règim permanent) per a cada nivell de tensió aplicat en borns del motor.

Per obtenir-la, s'ha realitzat un experiment consistent en variar progressivament la tensió d'alimentació i llegir la velocitat de l'eix mitjançant l'encoder. Les tensions es mouen en un rang de  $-21\text{V}$  a  $21\text{V}$  amb salts d' $1\text{V}$ . Als voltants de tensió nul·la (entre  $-4\text{V}$  i  $4\text{V}$ ) s'ha utilitzat una resolució superior per poder observar millor les no-linealitats de la planta. El nivell de tensió s'ha mantingut durant  $2\text{s}$ , temps suficient perquè s'arribi a l'estat estacionari, i s'han obviat les dades llegides durant el transitori. A més, l'experiment s'ha realitzat dues vegades, una incrementant la tensió de  $-21\text{V}$  a  $21\text{V}$  i una altra decrementant-la de  $21\text{V}$  a  $-21\text{V}$ .

La figura 4 mostra la corba estàtica obtinguda<sup>2</sup>. Com ja havíem suposat, al voltant de tensió nul·la, el sistema presenta una no-linealitat. Concretament, es pot observar una zona morta, és a dir, un rang de tensions d'entrada pels quals la velocitat de sortida és nul·la i fora de la qual la planta es comporta de forma lineal. Aquest fet es pot atribuir a l'existència del fregament sec, que provoca un parell resistent sobre l'eix del motor. A més, la figura mostra una histèresi, també atribuïble a aquest fregament sec, en ser el coeficient de fricció estàtic superior al dinàmic.

Això no obstant, fora d'aquesta zona, la dinàmica de la planta presenta una bona linealitat. Per tant, a l'hora d'identificar la part lineal del model, cal treballar en un rang de tensions d'entrada superior, aproximadament, a  $3.5\text{V}$  en valor absolut.

En aquest treball s'ha decidit realitzar el control del sistema per a valors positius de la velocitat, amb la qual cosa el model que s'identifiqui i el disseny del controlador són, en principi, tan sols adequats per a aquest rang de sortida. Per comprovar-ne la validesa per a velocitats negatives, caldria realitzar una anàlisi acurada de simetria en la resposta de la planta.

---

<sup>2</sup> L'script "plot\_c\_estat.m" que llegeix les dades experimentals i dibuixa la corba estàtica és a l'annex A.



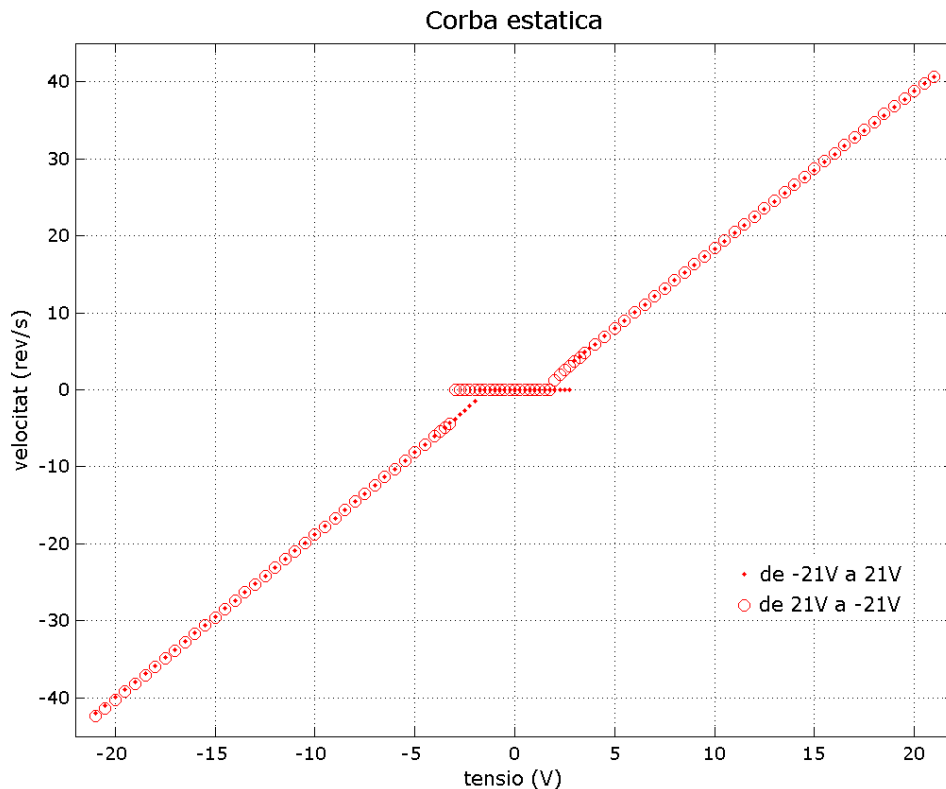


Figura 4: Corba estàtica de la planta

A partir del pendent de la corba estàtica en la zona lineal es pot calcular el guany de la planta, una dada important per a la determinació del model lineal. El valor del guany obtingut<sup>3</sup> per al rang de treball és de 2.0506.

### **3.4. Resposta temporal**

La resposta temporal permet caracteritzar la dinàmica del sistema. Particularment, permet fer hipòtesis sobre l'ordre del model a ajustar i la posició dels pols en llaç obert. Els senyals que solen utilitzar-se per a determinar la resposta temporal són senyals senzills, com impulsos, graons o rampes.

En el nostre cas, s'ha introduït un senyal quadrat de 5V d'amplitud i 4s de període. Com que es pretén treure informació de la part lineal del sistema, l'experiment s'ha de realitzar en la zona lineal de la corba estàtica. Per això, el senyal quadrat de l'entrada es mou en un rang de tensions de 4V a 9V. També cal tenir en compte que l'amplitud del graó d'entrada no pot ser molt gran, ja que podrien aparèixer comportaments no-lineals.

<sup>3</sup> La funció "sys\_gain.m" que llegeix les dades experimentals i calcula el guany de la planta és a l'annex A.

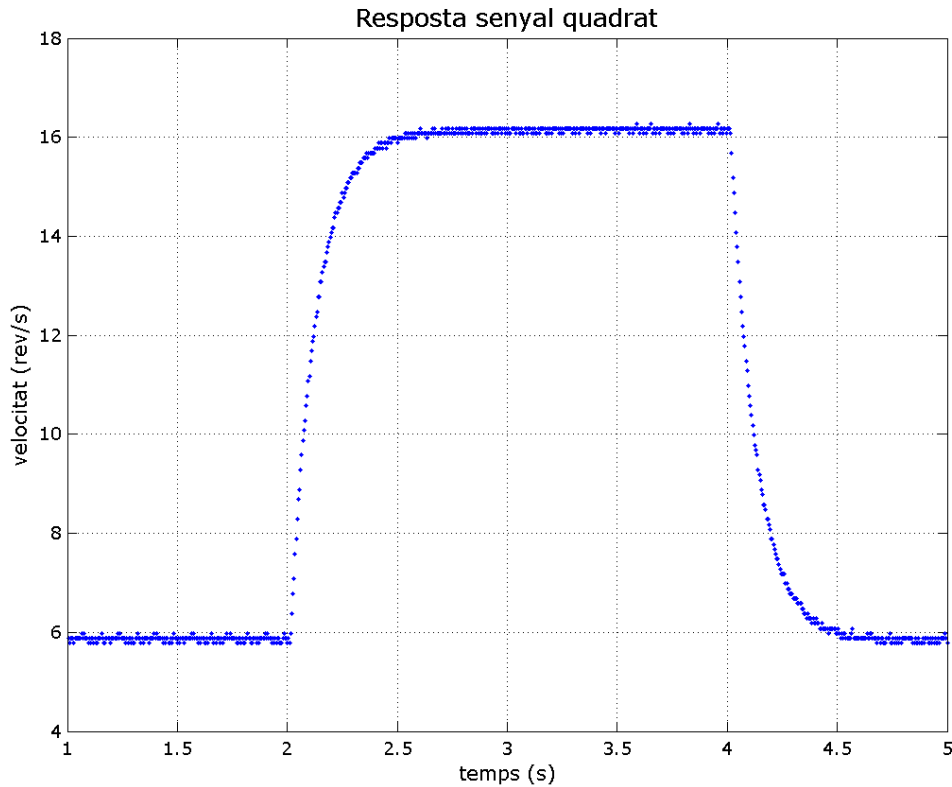


Figura 5: Resposta de la planta a un senyal quadrat

La figura 5 mostra les dades experimentals de velocitat llegides amb l'encoder per a un període del senyal d'entrada<sup>4</sup>. Es pot observar que és la resposta típica d'un sistema de primer ordre amb una constant de temps<sup>5</sup> de:

$$\tau \approx 0.1199s.$$

A més, a la figura 6, es pot observar que la resposta del sistema no és immediata, sinó que existeix un retard pur d'entre 10ms i 15ms. L'existència d'un retard pur és habitual quan s'utilitzen etapes PWM; tanmateix, també hi poden contribuir altres aspectes, com el software d'implementació.

En aquestes dues darreres figures també es pot observar que la sortida presenta una quantificació notable. De fet, aquesta quantificació està directament relacionada amb el període de mostreig  $T_s$ , de manera que, a menor període de mostreig, més efecte de la quantificació<sup>6</sup>.

<sup>4</sup> L'script "plot\_resp.m" que llegeix les dades experimentals i dibuixa la resposta temporal és a l'annex A.

<sup>5</sup> La funció "sys\_tau.m" que llegeix les dades i calcula la constant de temps és a l'annex A.

<sup>6</sup> Aquest fet deriva de l'expressió que s'utilitza per al càlcul de la velocitat:

$$\text{velocitat (rev/s)} = \frac{N \text{ polsos} \cdot 1 \text{ rev}/2000 \text{ polsos}}{T_s \text{ s}}$$

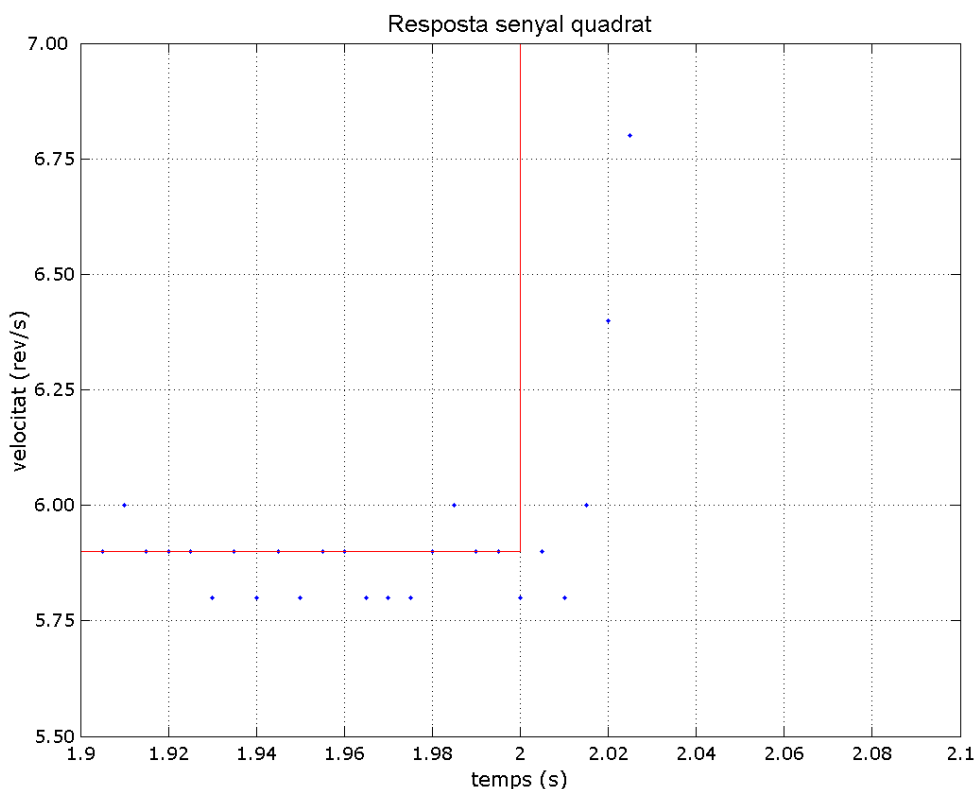


Figura 6: Detall de la resposta de la planta a un senyal quadrat en què s'observa el retard pur

### **3.5. Model lineal: Dynamic Signal Analyzer**

Un *dynamic signal analyzer* (DSA) és un aparell que permet obtenir el diagrama de Bode empíric d'un sistema, realitzant un escombrat en un rang de freqüències seleccionat. Per a cada iteració, l'aparell genera un sinus d'una determinada freqüència, l'entra a la planta i en llegeix el senyal de sortida. Comparant l'amplitud i calculant el desfasament entre el sinus generat i el primer harmònic del senyal de sortida de la planta, l'aparell pot dibuixar un punt en el diagrama de Bode (un per la corba de guany i un altre per la corba de fase).

Com que es pretén obtenir un model lineal, s'ha fet que el senyal sinusoidal d'entrada de la planta es moguéss entre 4V i 9V. S'ha realitzat un escombrat pràcticament des de contínua (50mHz) fins a 20Hz (de fet, el rang de freqüències en què es treballarà posteriorment i en el qual interessa tenir ben identificada la planta va des de 0Hz fins a uns 10Hz).

La figura 7 mostra l'espectre empíric obtingut amb el DSA. La corba de guany és la corba típica d'un sistema de primer ordre amb un pol situat a una freqüència d'entre 1Hz i 2Hz, a partir de la qual la corba baixa amb un pendent de

-20dB/dècada. La corba de fase, però, no és la característica d'un sistema de primer ordre en què el desfasament tendiria a  $-90^\circ$ . Això es deu a l'existència del retard pur, que no té influència en la corba de guany, però provoca un increment lineal del desfasament amb la freqüència, que en una escala logarítmica presenta l'aspecte de la figura. A partir de la corba de fase, doncs, es pot identificar aquest retard pur calculant-ne el pendent. S'ha de tenir en compte, tanmateix, que el càlcul del pendent s'ha de realitzar en aquelles freqüències on l'efecte del pol sobre el desfasament ja estigui exhaurit. Així, el valor del retard pur calculat és d'aproximadament 15.52ms.

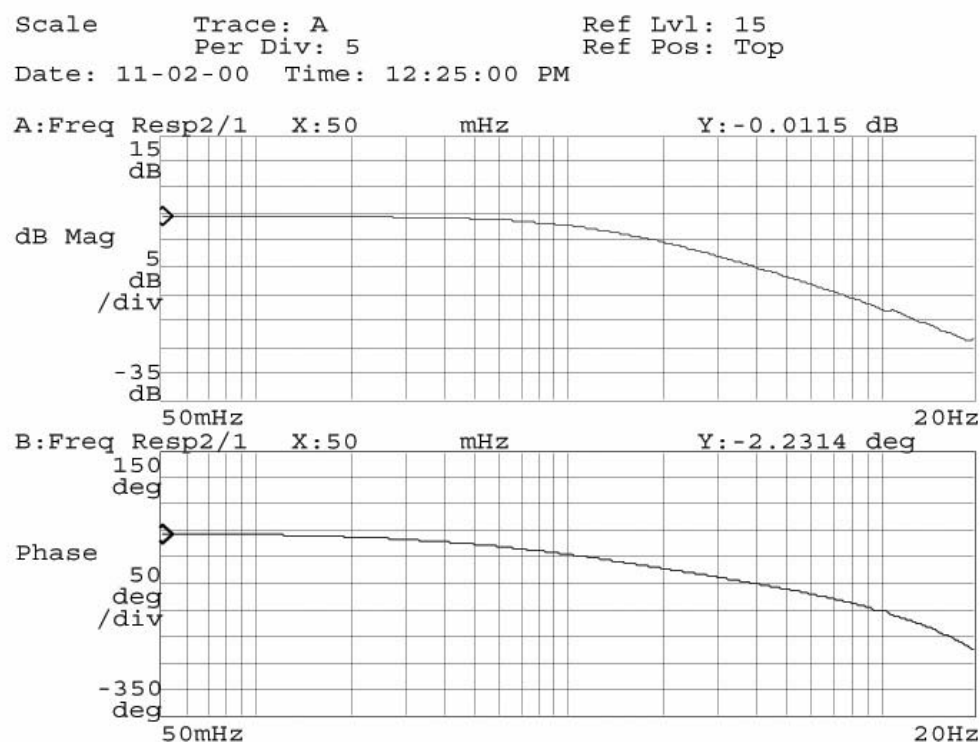


Figura 7: Diagrama de Bode empíric obtingut amb el DSA

El DSA, a més, també permet ajustar un model a partir del diagrama de Bode obtingut, proporcionant la posició dels pols i dels zeros del sistema. El model lineal ajustat és el que es presenta a continuació:

$$G_p(s) = \frac{16.5711}{s + 8.0810} e^{-15.52 \cdot 10^{-3} s}$$

La posició del pol ajustat correspon a una constant de temps  $\tau$  d'aproximadament 0.1237s, que és una mica més lenta que la constant de temps

calculada a partir de les dades experimentals. El valor del numerador s'ha calculat perquè el guany de la planta sigui de 2.0506.

### **3.6. Model lineal: System Identification Toolbox (MatLab)**

MatLab disposa d'una *toolbox* d'identificació de sistemes que permet ajustar models lineals en temps discret a partir de les dades reals d'entrada i sortida de la planta. També permet realitzar un post-processat de les dades d'entrada i sortida, com filtratge, eliminació de mitjanes, selecció de dades... A més, aquesta *toolbox* disposa d'una interfície gràfica (figura 8) que s'obre escrivint `ident` a la finestra de comandes de MatLab.

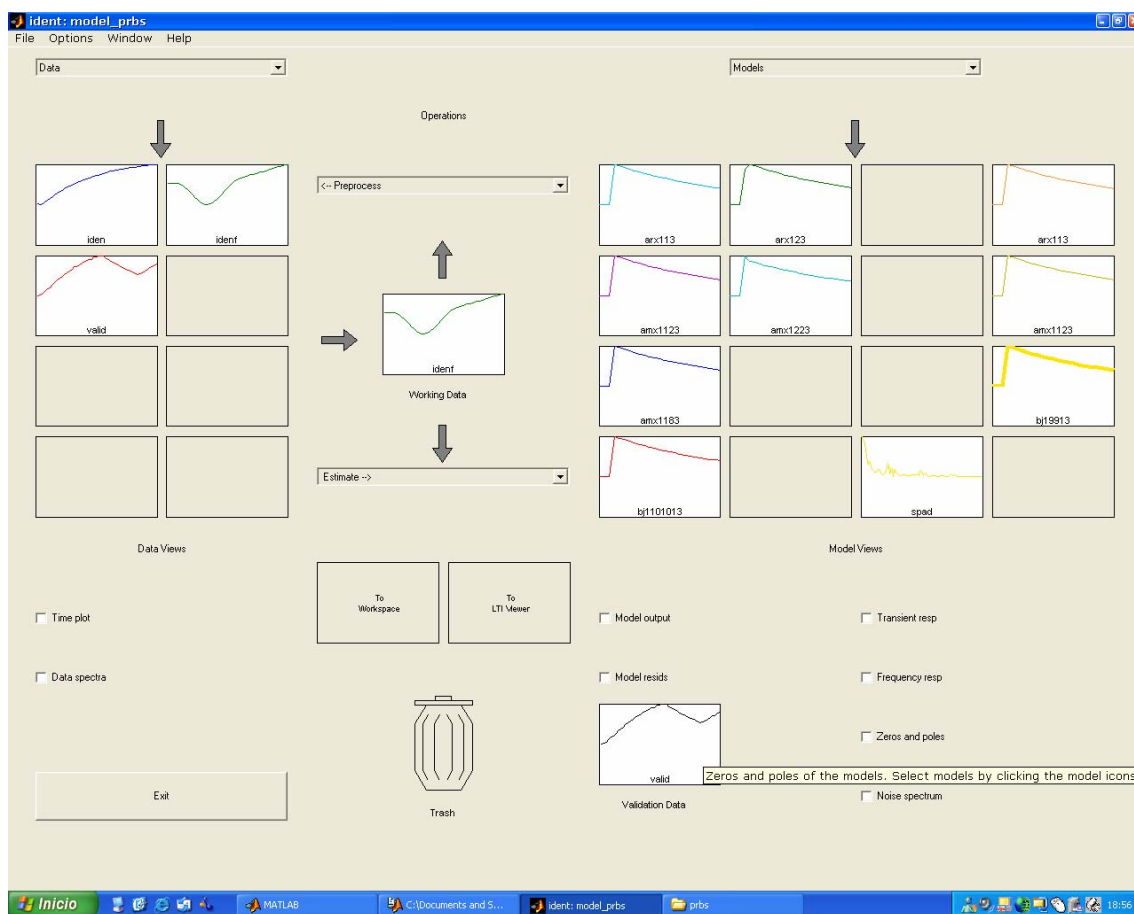


Figura 8: Interfície gràfica de la *toolbox*

Quant a l'ajust de models, la *toolbox* ofereix un ventall molt ampli de possibilitats: permet ajustar tant models no-paramètrics com models paramètrics, definir l'estructura del model, seleccionar-ne l'ordre, fixar el nombre de retards purs, de zeros... Una vegada ajustats diferents models, es té l'opció de comparar-

los i comprovar-ne la validesa amb d'altres dades experimentals. La *toolbox* proporciona la resposta temporal (per entrada graó) del model estimat, la resposta freqüencial, l'ajust (*fit*) a altres dades experimentals de validació, la posició dels pols i dels zeros i una anàlisi de residus. Posteriorment, el model estimat es pot exportar al *workspace* i manipular-lo com una variable més per a altres estudis o simulacions.

Així doncs, per poder aprofitar les prestacions d'aquesta *toolbox*, ha estat necessari realitzar un experiment que proporcionés les dades d'entrada i sortida.

En tasques d'identificació de sistemes és de vital importància el senyal que s'utilitza per excitar el sistema d'estudi. Concretament, interessa que el senyal sigui prou ric per excitar el sistema en el rang de freqüències en què ens interessa identificar el model. En aquest treball s'ha optat per utilitzar les Seqüències Binàries Pseudo-Aleatòries (PRBS<sup>7</sup>). Aquests senyals prenen únicament dos valors i, per tant, tenen una amplitud acotada; tenen un espectre de potència més ric que d'altres senyals d'excitació, com els senyals quadrats; i el seu ample de banda pot escollir-se fàcilment per cobrir l'ample de banda del sistema d'estudi.

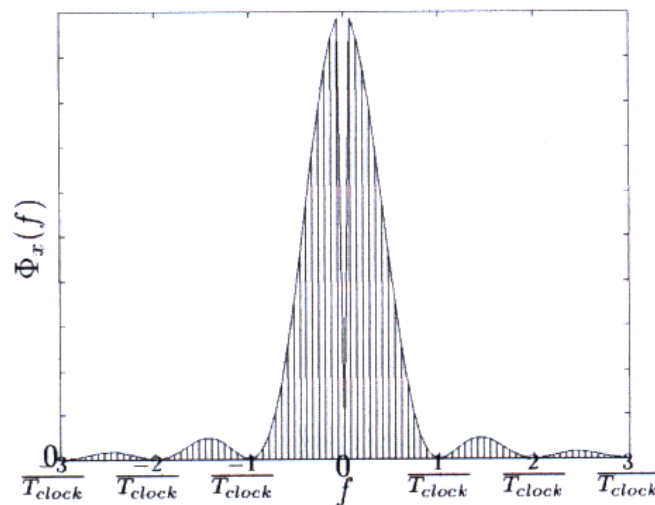


Figura 9: Densitat espectral de potència d'un senyal PRBS

La figura 9 mostra la densitat espectral de potència del senyal PRBS. A efectes pràctics, l'espectre del senyal PRBS pot considerar-se pràcticament pla fins a un valor de  $0.4(1/T_{\text{clock}})$  i, per tant, a aquesta freqüència s'haurien de cobrir tots els modes rellevants del sistema.

<sup>7</sup> De l'anglès, *Pseudo-Random Binary Sequences*.

El MatLab i, en concret, la *toolbox* d'identificació, permet generar senyals d'excitació mitjançant al comana:

```
u = idinput(N, type, band, levels);
```

Per generar un senyals PRBS, cal seleccionar el *type* 'prbs'; *N* indica la longitud de la seqüència PRBS; *band*, el seu ample de banda; i *levels*, els dos nivells de sortida. Al MatLab, però, cal introduir *band* com un vector de dues components  $[0, 1/B]$ , on  $1/B$  equival a  $T_s/T_{\text{clock}}$ , sent  $T_s$  l'interval de temps entre mostres successives del senyal.

Per tant, abans de generar el senyal PRBS amb el MatLab cal decidir el període de mostratge que s'utilitza per realitzar l'experiment amb la planta real. Amb el criteri d'obtenir, com a mínim, 40 mostres al transitori, es pren un període de mostratge de 5ms.

En el nostre cas, com ja s'ha comentat anteriorment, interessa excitar el sistema des de contínua fins a aproximadament 10Hz. Per tant:

$$1/T_{\text{clock}} = 25\text{Hz}$$

$$1/B = 0.125$$

Així doncs, la comana que genera la seqüència PRBS és:

```
u = idinput(2048, 'prbs', [0, .125], [-1,1]);
```

La figura 10 mostra l'espectre<sup>8</sup> del senyal PRBS generat. Com es pot observar, aquest presenta una corba aproximadament plana des dels 100mHz fins als 10Hz. Tanmateix, per a la nostra aplicació, interessa tenir molt ben identificada la contínua, i aquest senyal PRBS generat no garanteix que el sistema arribi al règim permanent en cap instant de temps. Per això, s'ha afegit un senyal quadrat davant del senyal PRBS d'un període prou llarg com perquè la resposta arribi a l'estat estacionari. La figura 11 mostra l'espectre del nou senyal comparat amb el del senyal anterior. Com es pot observar, ara la corba excita des dels 50mHz fins als 10Hz, que és el mateix rang que s'ha utilitzat en l'identificació amb el DSA.

La figura 12 mostra el senyal d'excitació que s'introdueix a la planta, amb el senyal quadrat seguit del senyal PRBS generat amb MatLab.

Per realitzar l'experiment, s'exporta aquest senyal al software de control i, mostrejat periòdicament cada 5ms, s'introdueix a la planta. Com que es vol identificar la part lineal del model del sistema, els dos nivells d'aquest senyal es passen, a nivell de software, a unes tensions equivalents de 4V i 9V. La velocitat de sortida es calcula a partir dels senyals de l'encoder.

---

<sup>8</sup> L'script "espectre\_prbs.m" que dibuixa l'espectre del senyal PRBS és a l'annex A.

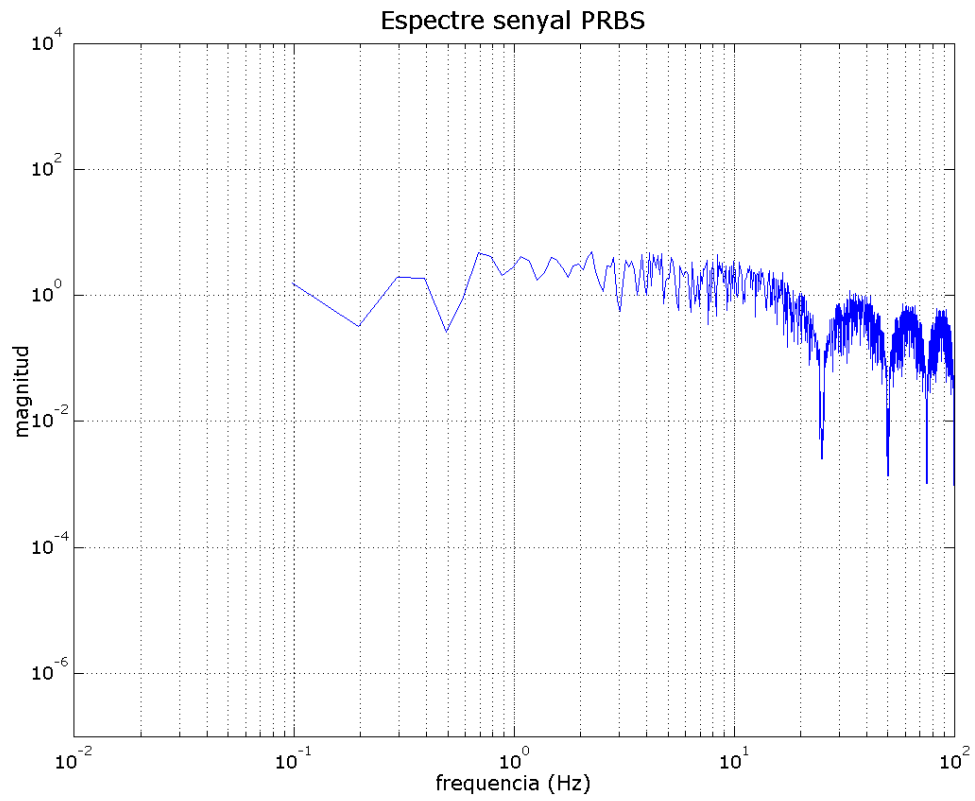


Figura 10: Espectre del senyal PRBS generat amb MatLab

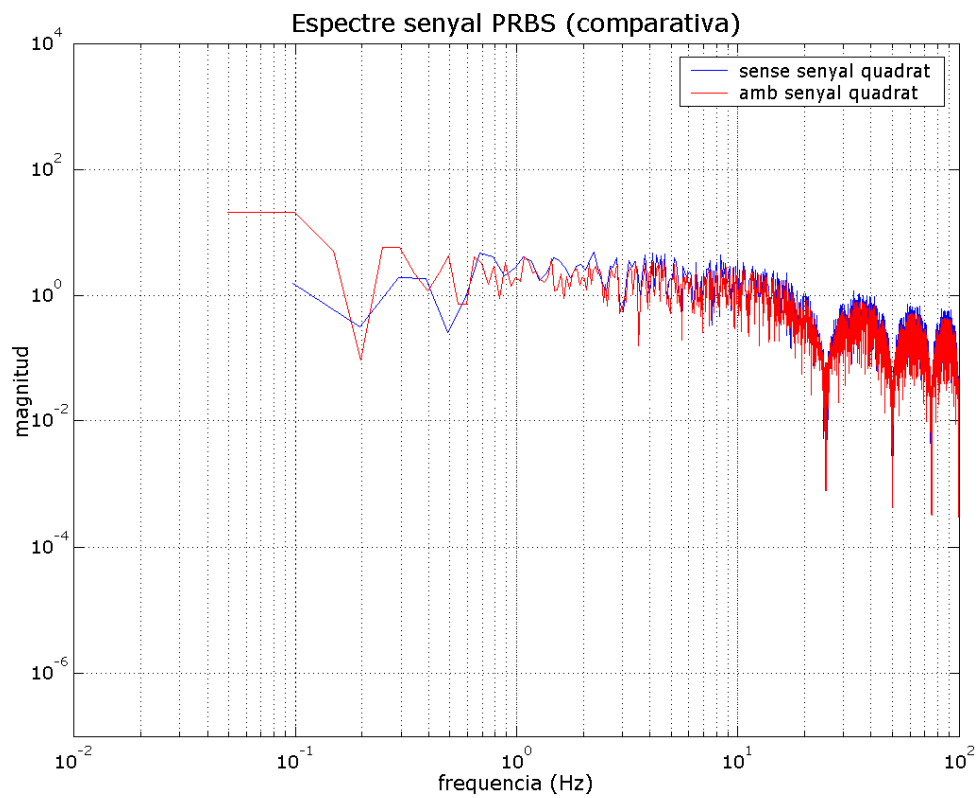


Figura 11: Comparativa dels espectres dels dos senyals d'excitació



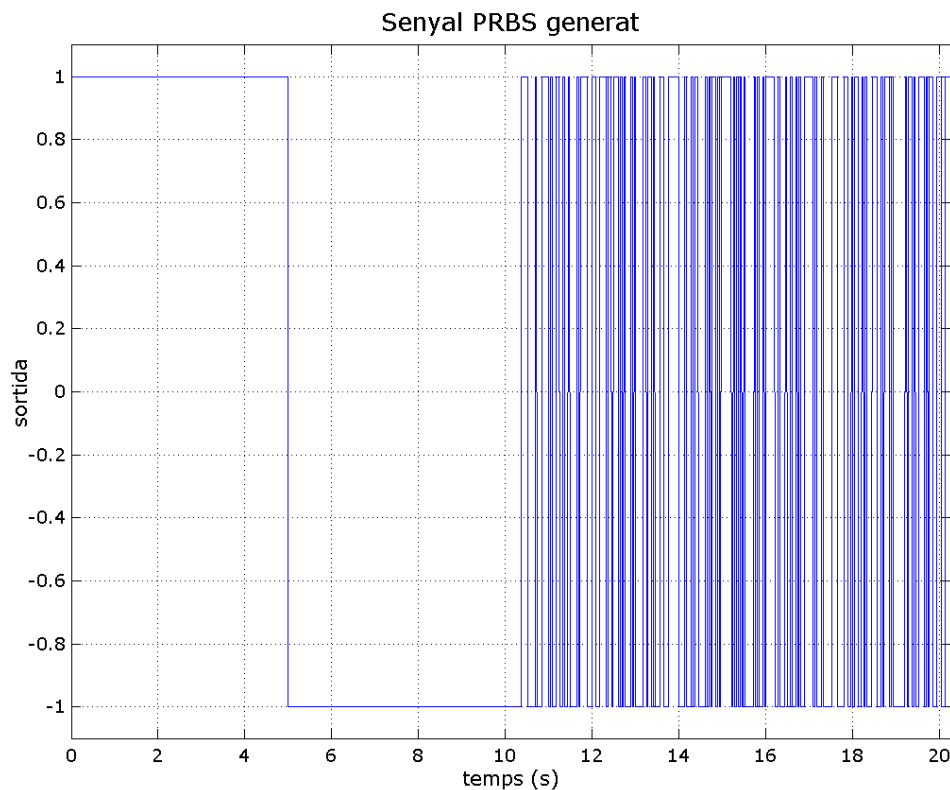


Figura 12: Senyal d'excitació per a la identificació de la planta.

La figura 13 mostra un detall de la resposta del sistema al senyal PRBS d'entrada. Els valors obtinguts es podrien introduir, ara ja, a la *toolbox* d'identificació de sistemes. Tanmateix, per facilitar la feina de la *toolbox*, se centren i s'escalen tant el senyal d'entrada com el de sortida de la planta en un rang de  $-1$  a  $1$ . Aquest tractament es podria realitzar amb la pròpia *toolbox*, però s'ha preferit controlar aquest procés d'adequació de senyals<sup>9</sup>.

És important destacar que s'han realitzat dos experiments amb dos senyals d'excitació diferents però amb el mateix ample de banda. La *toolbox* permet utilitzar unes dades per a l'identificació del model lineal i unes altres per a la validació dels resultats. Això evita que la *toolbox* "aprengui" les dades d'identificació, de manera que el model només ajusti bé aquestes dades.

Abans de començar a ajustar models lineals, cal comprovar que té sentit ajustar-los, és a dir, que el conjunt de dades d'entrada i sortida de la planta presenten una certa linealitat. La coherència espectral és un bon indicador de la linealitat d'un sistema. Indica, per a cada freqüència, quina energia del senyal

<sup>9</sup> La funció "prbs\_adapt.m" que llegeix les dades experimentals i realitza el tractament dels senyals és a l'annex A.

d'entrada passa al senyal de sortida amb la mateixa freqüència. Per tant, és funció de la freqüència i pren valors entre 0 i 1. A la pràctica, s'accepta la linealitat del sistema per aquelles freqüències en què la coherència és superior a 0.9.

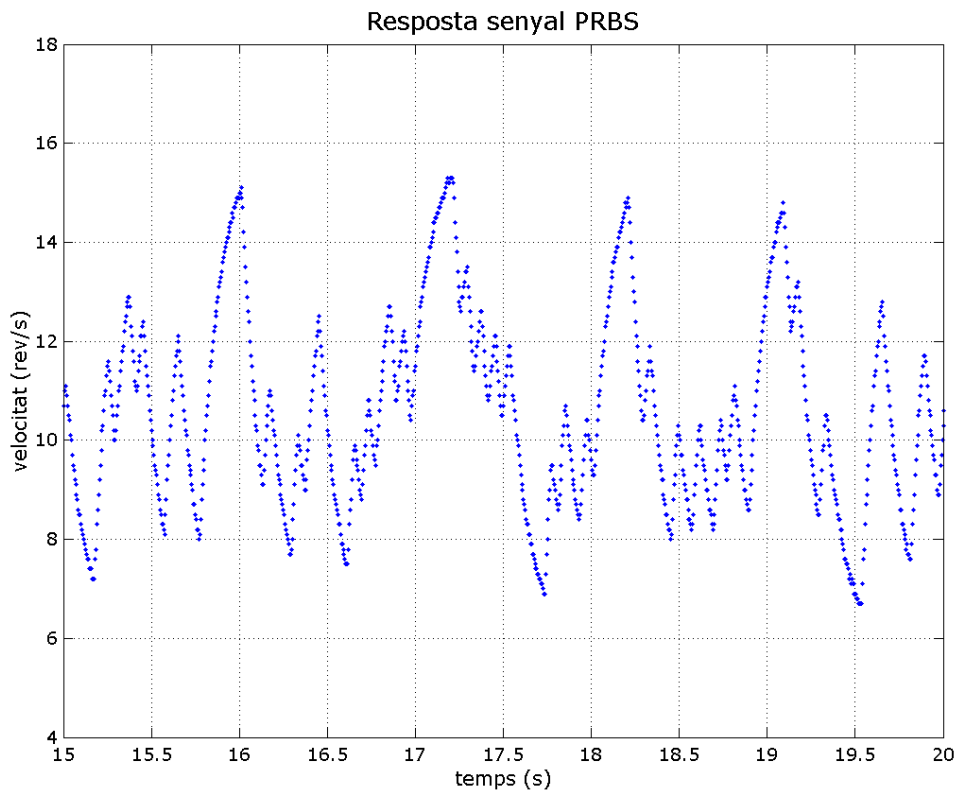


Figura 13: Detall de la resposta al senyal PRBS d'excitació.

La comana de MatLab que permet dibuixar la coherència espectral de dos senyals  $x$  i  $y$  és la següent:

```
cohere(x, y, NFFT, Fs);
```

on  $NFFT$  és la longitud de la DFT; i  $Fs$ , la freqüència de mostratge. En el nostre cas, la comana és, amb la notació utilitzada en les funcions:

```
cohere(iden.voltage, iden.velo, 512, 200);
```

La figura 14 mostra la coherència espectral dels senyals d'entrada i sortida obtinguts. Com es pot observar, la coherència és superior a 0.9 només per a freqüències inferiors als 20Hz. Per tant, el model lineal que s'obtingui a partir d'aquestes dades experimentals només té sentit per a aquest rang de freqüències. Com que coincideix amb el rang en què interessa identificar correctament la dinàmica de la planta es pot utilitzar la *toolbox* d'identificació.

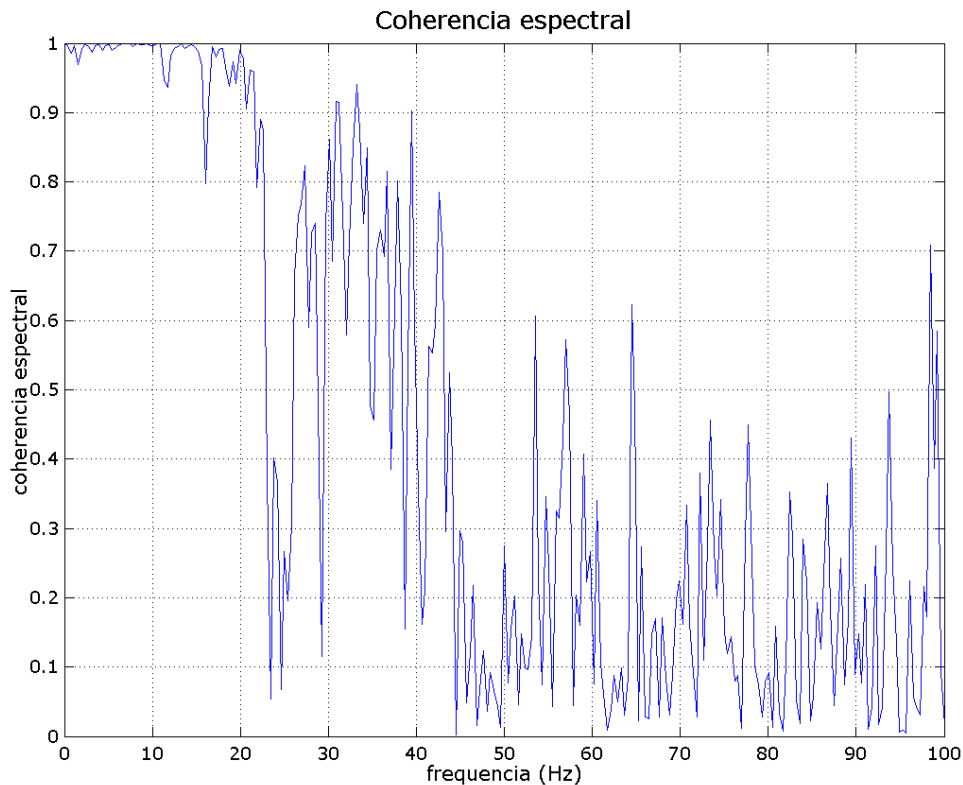


Figura 14: Coherència espectral dels senyals d'entrada i sortida de la planta.

L'arxiu "models\_ajust.sid" obre la sessió de la *toolbox* d'identificació en què s'han ajustat els diferents models lineals. Per a l'estimació del model s'han utilitzat dades d'entrada i sortida diferents que per a la validació del model. Aprofitant les eines de processat de dades de la *toolbox*, les dades d'estimació s'han filtrat amb un filtre passa-baixos amb una banda passant de 0Hz a 10Hz per eliminar el soroll de quantificació de l'encoder en el senyal de sortida. Així, el model que millor ajusta la resposta del sistema és un Box-Jenkins(19913) i la seva funció de transferència en temps discret és:

$$G_p(z) = \frac{0.0900}{z - 0.9561} z^{-3} \quad (T_s = 0.005s)$$

on el valor del numerador s'ha calculat perquè el guany de la planta sigui de 2.0506.

La figura 15 mostra un detall de l'ajust que el model estimat fa de les dades experimentals de validació. És un dels resultats que facilita la *toolbox* i que permet veure, de forma qualitativa, si el model ajusta bé la resposta del sistema. La línia negra són els valors reals i la blava la resposta del sistema estimat. Com es pot observar, la resposta s'assembla força a les dades reals. Les diferències més

significatives s'observen als pics, on existeixen components d'alta freqüència que el model no ajusta bé. Aquesta gràfica, però, no ens permet decidir a simple vista si el model explica prou bé la dinàmica del sistema o es deixa d'explicar-ne alguna part important.

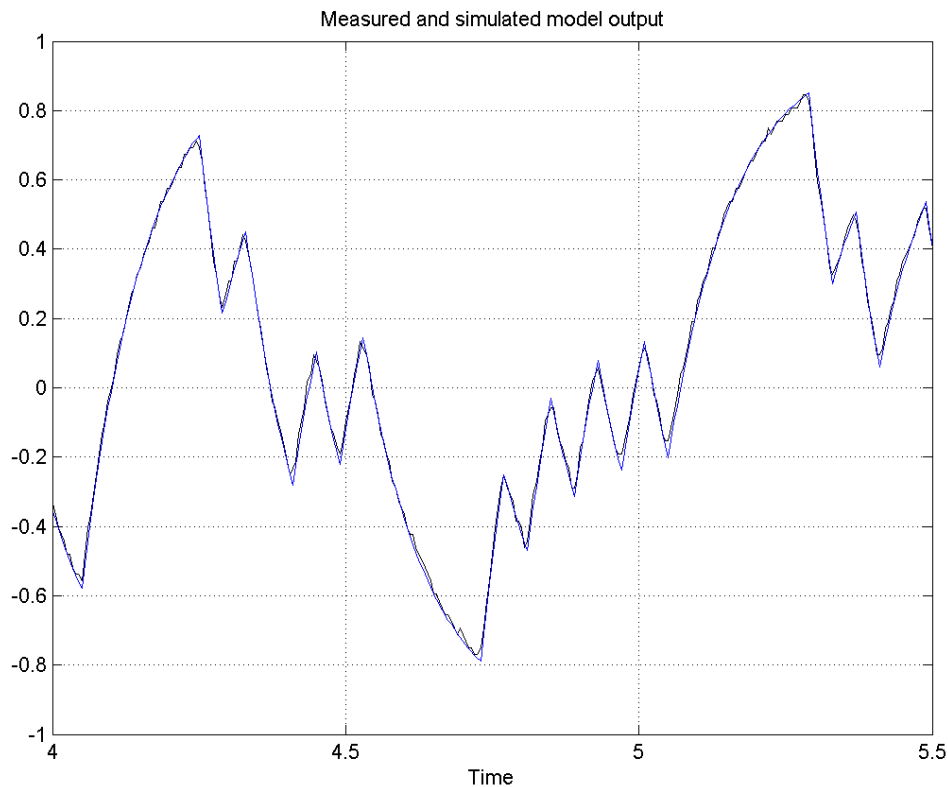


Figura 15: Comparativa de les dades experimentals i de la resposta del model estimat per la *toolbox*.

Una bona manera de comprovar-ho és a partir de l'anàlisi de residus que proporciona la *toolbox* (figura 16). Les línies discontinúes indiquen l'interval de tolerància del 95%. Es pot observar que la correlació creuada entre l'entrada i els residus de la sortida es manté dins els marges, mentre que l'autocorrelació dels residus de la sortida surt dels marges i presenta una marcada estructuració, la qual cosa significa que el model no explica una part de la dinàmica del sistema. Tanmateix, es probable, tenint en compte la geometria de l'estructuració, que la part que no explica el model sigui el soroll de la quantificació de l'encoder. Així doncs, considerem com a bo el model obtingut.

Tot i que, en aquest cas, l'estructura del model ajustat amb la *toolbox* coincideix amb el model obtingut empíricament a partir del DSA, s'observa el risc d'ajustar models d'ordres excessivament elevats (i, per tant, amb més paràmetres), ja que la *toolbox* no penalitza els models d'ordres elevats, sinó que ha

de ser el propi usuari que realitzi aquesta funció, analitzant si l'augment de la part explicada de les dades compensa l'increment d'ordre del model.

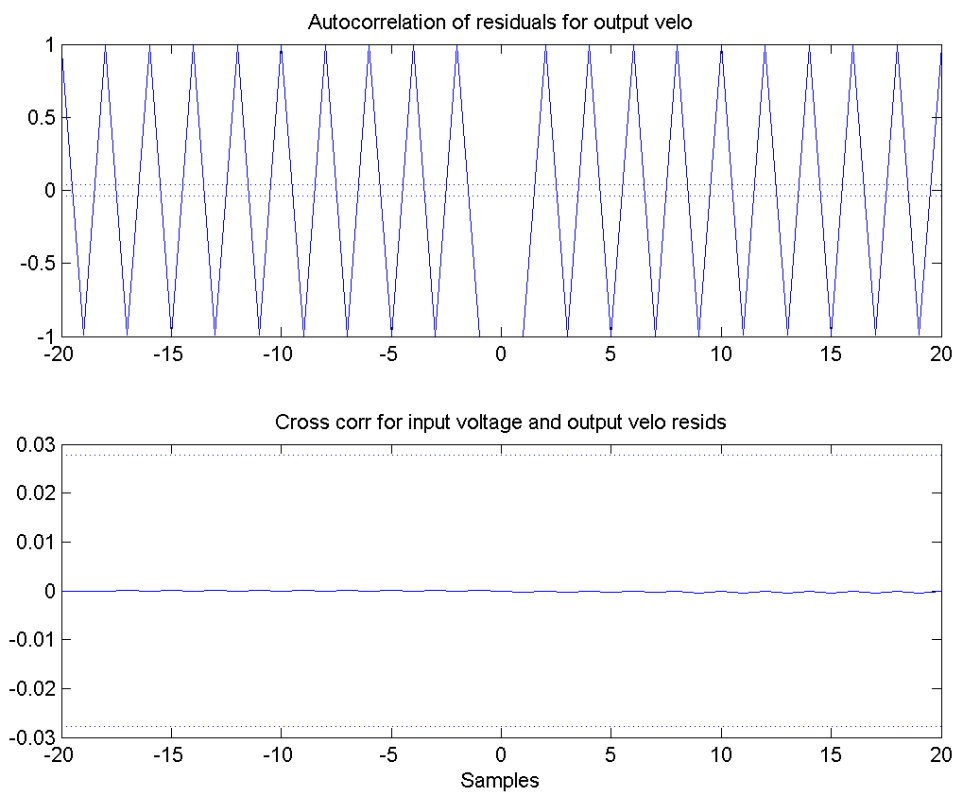


Figura 16: Anàlisi dels residus del model ajustat proporcionat per la *toolbox*.

### **3.7. Model lineal: equacions físiques**

Com ja s'ha comentat, el modelat d'un sistema mitjançant les equacions físiques permet entendre el comportament del sistema i determinar les causes dels efectes que es poden observar durant l'experimentació.

Aquesta tècnica, en general, requereix un conjunt de supòsits simplificatius. Per tant, cal tenir present sempre que el modelat d'un sistema representa la realitat tan sols sota un conjunt molt limitat de condicions experimentals i que, per a obtenir resultats satisfactoris, mai no s'ha de perdre de vista el comportament real del sistema. Les simulacions ens ajuden a comprovar la validesa d'aquests supòsits, contrastant-ne els resultats amb la pràctica.

Com ja s'ha comentat, la planta està formada per un motor de corrent continu al qual s'ha acoblat un dispositiu magnètic, que provoca un parell variable sobre l'eix del motor a mode de pertorbació. Cal recordar que, en aquest treball, de

moment, només interessa identificar el model dinàmic del motor, sense tenir en compte l'efecte del dispositiu magnètic.

A continuació s'indiquen les equacions diferencials que descriuen el comportament del motor. A [1] s'explica detalladament la deducció d'aquestes equacions:

- part elèctrica

$$V_a - I \cdot R - L \cdot \frac{dI}{dt} - k_v \cdot \frac{d\theta}{dt} = 0$$

on  $V_a$  és el voltatge aplicat al motor en Volts;  $I$  és el corrent que circula pel rotor en Ampers;  $R$  és la resistència dinàmica del motor en Ohms;  $L$  és el coeficient d'autoinducció de la bobina en Henry;  $k_v$  és la constant de velocitat en V/s; i  $d\theta/dt$  és la velocitat angular de l'eix del motor en  $s^{-1}$ .

- part mecànica

$$J \cdot \frac{d^2\theta}{dt^2} = k_t \cdot I + \Gamma_L - B \cdot \frac{d\theta}{dt}$$

on  $J$  és el moment d'inèrcia del sistema respecte l'eix del motor en  $kg \cdot m^2/s^2$ ;  $k_t$  és la constant de parell en  $N \cdot m/A$ ;  $\Gamma_L$  és el parell de càrrega en  $N \cdot m$ ; i  $B$  és el coeficient de fregament viscos en  $N \cdot m \cdot s$ .

En el nostre cas, el motor gira en buit i, per tant, el parell de càrrega  $\Gamma_L$  és nul. La figura 17 mostra el diagrama de blocs del sistema.

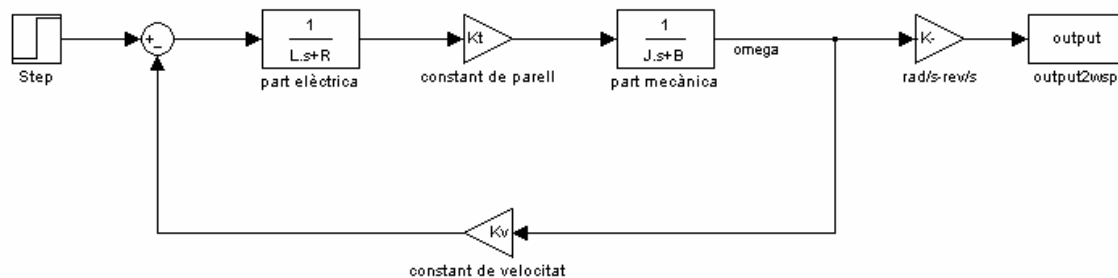


Figura 17: Diagrama de blocs del sistema en laïç obert (model Simulink).

Per acabar l'identificació cal realitzar un ajust dels paràmetres del model. El procediment que s'ha seguit és el que s'explica a [2], aprofitant els valors calculats

anteriorment per la constant de temps i el guany canònic del sistema. Així, els valors obtinguts<sup>10</sup> són:

$J = 1.2870e-005;$	en $\text{kg}\cdot\text{m}^2/\text{s}^2$
$B = 3.5610e-005;$	en $\text{N}\cdot\text{m}\cdot\text{s}$
$L = 1.4850e-002;$	en H i a 1kHz (freqüència del PWM: 1.2kHz)
$R = 31.6400;$	en $\Omega$ i a 1kHz (freqüència del PWM: 1.2kHz)
$K_v = 5.8280e-002;$	en V/s
$K_t = 5.8280e-002;$	en $\text{N}\cdot\text{m}/\text{A}$

Aquest model, a diferència dels anteriors, és de segon ordre i no presenta cap retard pur. Quant a l'ordre, el pol associat a la part elèctrica és molt ràpid (de manera que domina el pol associat a la part mecànica), i es troba a altes freqüències (per tant, lluny del rang d'interès), la qual cosa explica que no hagi estat identificat pels altres dos mètodes. Pel que fa al retard pur, és probable que provingui de l'etapa PWM i del software, elements que no han estat tinguts en compte en el desenvolupament del model.

### **3.8. Model complet**

Ja s'han identificat les no-linealitats de la planta i s'han obtingut tres models lineals que descriuen la dinàmica de la planta en la zona lineal. Ara cal, doncs, unir la part no-lineal amb la part lineal i obtenir un model complet que descrigui el comportament del sistema en totes les condicions de funcionament.

Com ja s'ha comentat, aquests models han de servir, d'una banda, per al disseny del controlador i, d'altra, per a la realització de simulacions per validar l'algorisme de control. En els casos en què es disposa de diversos models del sistema d'estudi, és usual realitzar les simulacions contra un model diferent del que s'utilitza per al disseny del controlador.

En aquest treball s'ha decidit utilitzar el model obtingut a partir de les equacions físiques per realitzar les simulacions, principalment perquè, com ja es veurà més endavant, es disposa d'un model, també basat en lleis físiques, per generar les pertorbacions del dispositiu magnètic. A més, el fet que les no-linealitats tinguin una explicació física (fregament sec) permet incloure-les en el punt lògic del model lineal, com a suma d'un parell resistent.

---

<sup>10</sup> Els paràmetres estan guardats a l'arxiu "wsp\_planta.mat" a la carpeta \model\_físic\planta del CD adjunt.

En canvi, per al disseny del controlador, s'han utilitzat els altres dos models, l'obtingut a partir del DSA i l'obtingut a partir de la *toolbox* de MatLab.

Per tant, el model que ha de descriure de la millor manera possible el comportament de la planta, incloses les no linealitats, ha de ser el model de les equacions físiques. Dels altres dos, només en cal comprovar que ajusten prou bé la resposta temporal, sense donar especial importància a la part no-lineal.

### Model basat en lleis físiques

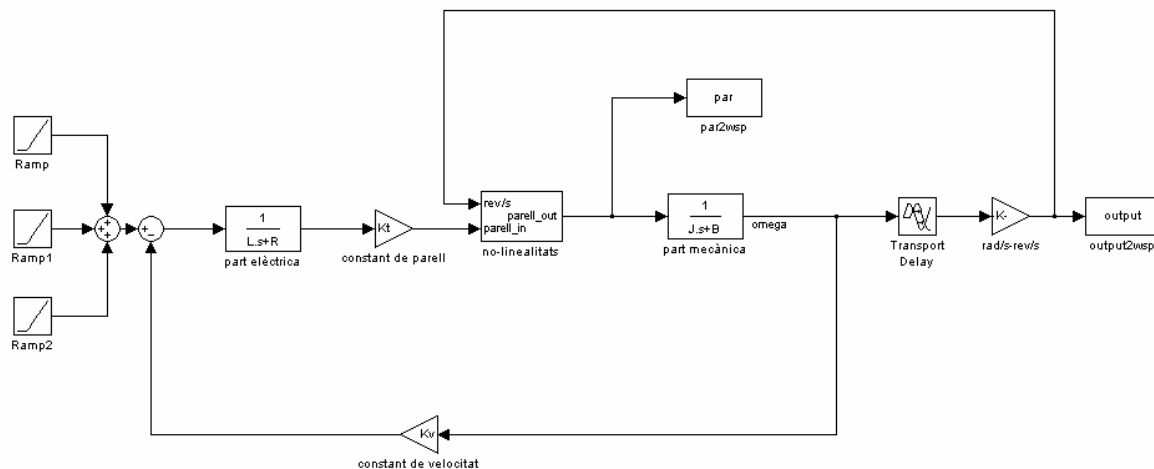


Figura 18: Diagrama de blocs del sistema enllaç obert amb les no-linealitats (model Simulink).

La figura 18 mostra com queda el diagrama de blocs del sistema amb la inclusió de les no-linealitats<sup>11</sup> i del retard pur. Com ja s'ha comentat, aquest bloc simula l'efecte del fregament sec, tant la zona morta com la histèresi. Com es pot observar a la figura, el bloc necessita la lectura de la velocitat; cal remarcar que no es tracta, en cap cas, d'una realimentació. L'ajust dels valors de la part no-lineal s'ha realitzat de forma que la resposta del sistema quadrés amb els resultats obtinguts en la corba característica obtinguda prèviament.

La figura 19 mostra la resposta temporal simulada del model Simulink obtingut per a un període d'una ona triangular de 5V d'amplitud i 10s de període. Es poden observar clarament els efectes de la zona morta i de la histèresi.

<sup>11</sup> La composició del bloc que modela les no-linealitats es pot analitzar amb més detall a l'arxiu "planta.mdl", a la carpeta \model\_físic\planta\ del CD adjunt. Per realitzar simulacions cal carregar el *workspace* associat "wsp\_planta.mat".



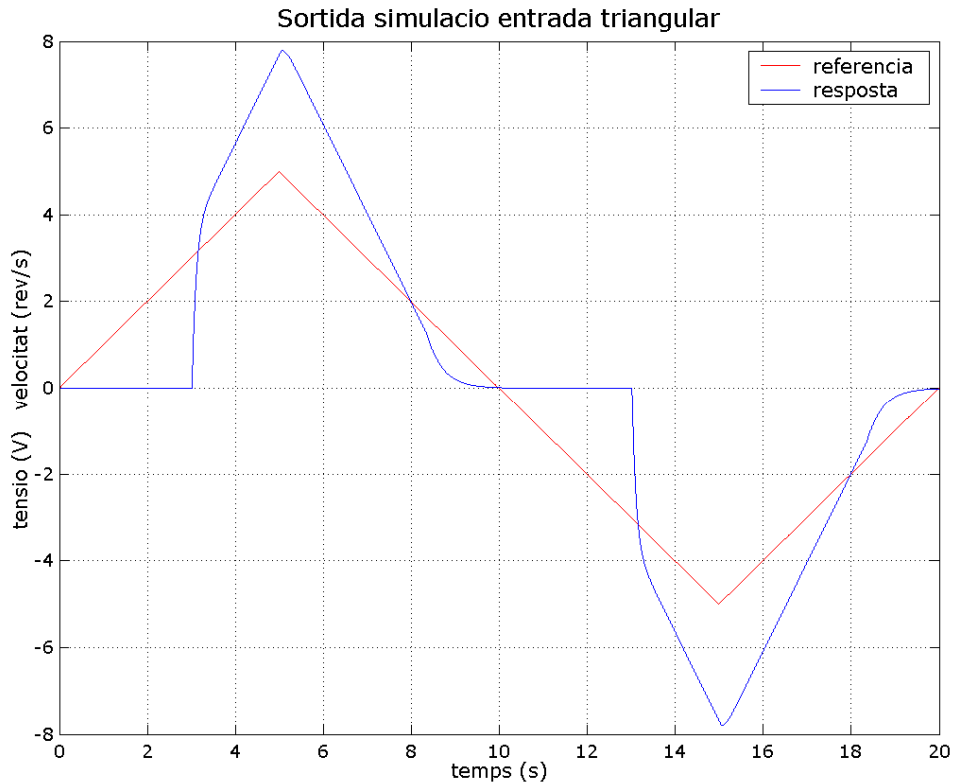


Figura 19: Resposta simulada del model ajustat per a una ona triangular a l'entrada.

#### Models obtinguts amb el DSA i la *toolbox*

La figura 20 mostra el mapa de pols i zeros (*pzmap*) dels dos models ajustats. Els pols es troben pràcticament al mateix lloc, la qual cosa significa que ambdós models tenen un comportament dinàmic semblant. El fet que els dos models, ajustats per procediments diferents, siguin tan semblants és un bon senyal i, per tant, probablement descriuen correctament la dinàmica del sistema dins el rang de freqüències d'interès.

La figura 21 mostra la resposta al graó simulada dels dos models<sup>12</sup> solapada a les dades experimentals obtingudes. Com es pot observar, el model obtingut a partir de la *toolbox* ajusta millor les dades i, per tant, en aquest treball s'ha utilitzat aquest model per al disseny del controlador. Tanmateix, a la pràctica, seria interessant provar els dos models per al disseny i observar quin es comporta millor contra la planta real.

<sup>12</sup> Els arxius "sim\_dsa.mdl" i "sim\_toolbox.mdl" permeten realitzar simulacions amb els dos models obtinguts. Aquests dos models només tenen sentit per a tensions d'entrada i velocitats de sortida positives. Els arxius es troben, respectivament, a les carpetes \model\_dsa\planta\ i \model\_toolbox\planta\.

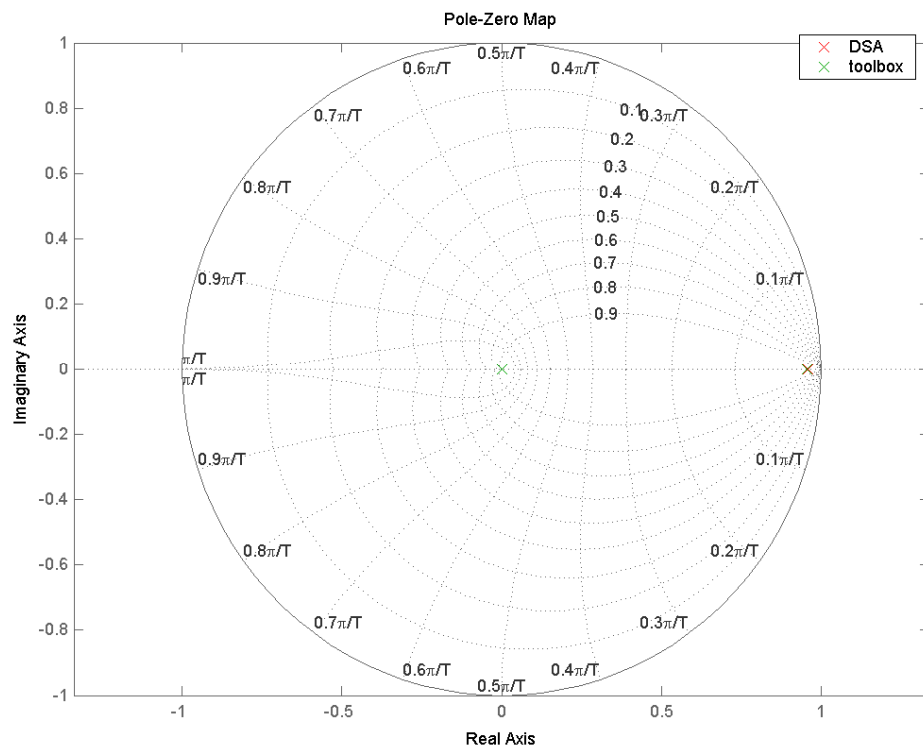


Figura 20: Mapa de pols i zeros dels dos models ajustats.

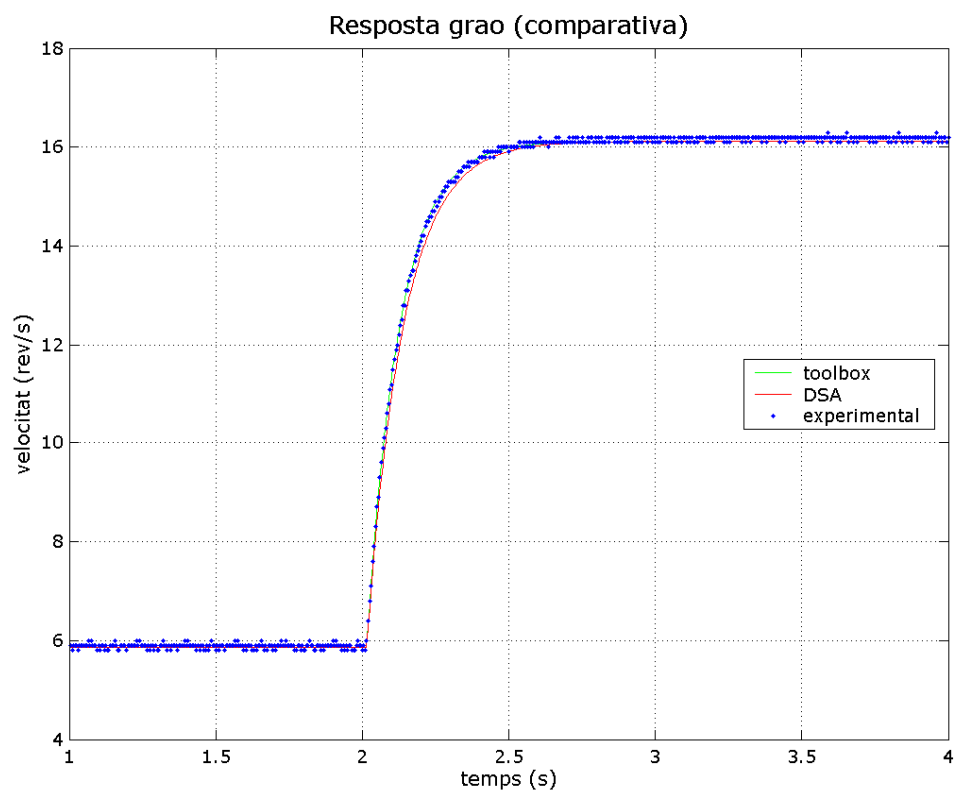


Figura 21: Resposta al graó dels dos models ajustats.

Per dibuixar el mapa de pols i zeros i per realitzar les simulacions s'ha passat el model en temps continu obtingut amb el DSA a un model en temps discret per a un període de mostratge de 5ms. Les comandes MatLab per a realitzar aquesta transformació són les següents:

```
% periode de mostratge
Ts=.005;
% model DSA
nGp=16.5711;
dGp=[1, 8.0810];
Gp=tf(nGp,dGp);
Gp_D=c2d(Gp,Ts);
% afegim tres retards purs (15ms~15.52ms)
num=1;
delays=3;
den=zeros(1,delays+1);
den(1)=1;
Gret=tf(num,den,Ts);
% funcio de transferencia en temps discret
Gp_D=Gp_D*Gret;
```

## **Capítol 4**

# **CONTROL REPETITIU**

### **4.1. Introducció**

Existeixen moltes aplicacions en què es fa necessari el seguiment d'una consigna periòdica o el rebuig d'una pertorbació periòdica que altera la sortida d'un sistema. Aquestes qüestions no es poden resoldre a partir dels mètodes de control clàssics com els controladors PID. Cal recórrer a d'altres tècniques.

El control repetitiu és una tècnica de control lineal que utilitza el Principi del Model Intern [3] i que es mostra especialment eficient per a resoldre tant el seguiment de consignes periòdiques com el rebuig de pertorbacions periòdiques.

Per realitzar aquest seguiment/rebuig en llaç tancat sense error en estat estacionari, el controlador introdueix un guany infinit (o molt gran) a la freqüència fonamental del senyal de referència/pertorbació i als seus harmònics<sup>13</sup>.

Una restricció important en el disseny del controlador és que el període de mostratge del controlador ha d'anar sincronitzat amb el període de referència o de la pertorbació, de manera que s'obtingui un nombre enter de mostres per període.

La figura 22 mostra el diagrama de blocs del sistema de llaç tancat  $G_c(z)$  és el controlador nominal i estabilitza la planta,  $G_p(z)$ , en llaç tancat. Aquest esquema també introdueix un filtre  $G_x(z)$  per assegurar l'estabilitat del llaç tancat i un filtre  $H(z)$  passabaixos que serveix per a donar robustesa al controlador.

A [4] s'explica amb una mica més de detall la Teoria del Control Repetitiu i es descriu el procediment de disseny dels diferents blocs per a garantir l'estabilitat del sistema en llaç tancat.

---

<sup>13</sup> Cal tenir en compte que, en la implementació en temps discret, tan sols es podran seguir amb un error nul en estat estacionari aquells harmònics que es trobin per sota de la freqüència de Nyquist. Per tant, el període de mostreig  $T_s$  s'ha d'escollir de manera que totes les components de freqüència que es vulguin seguir/rebutjar estiguin per sota de la freqüència de Nyquist.

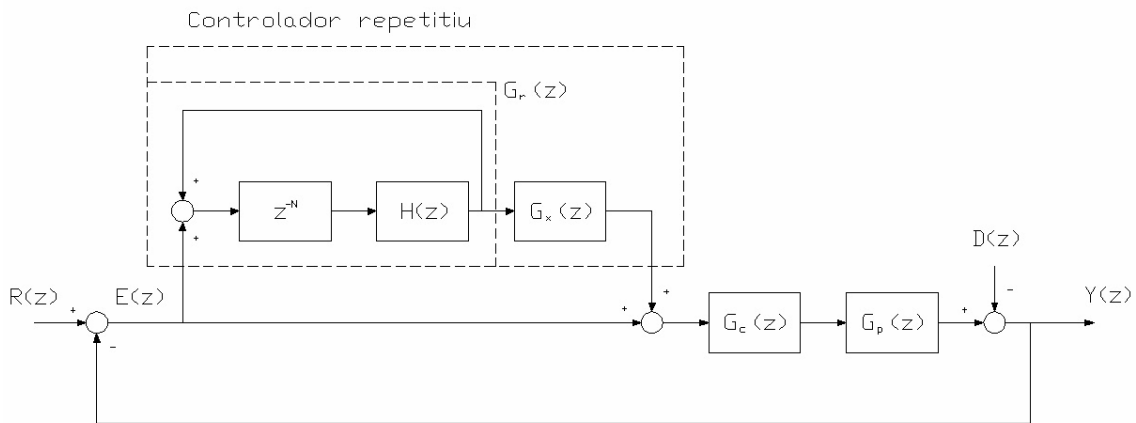


Figura 22: Diagrama teòric de blocs del sistema de control.

#### 4.2. Model de la pertorbació

Abans de començar a dissenyar el controlador, es presenta un model basat en lleis físiques del dispositiu magnètic que genera les pertorbacions. El desenvolupament teòric d'aquest model es pot trobar a [1].

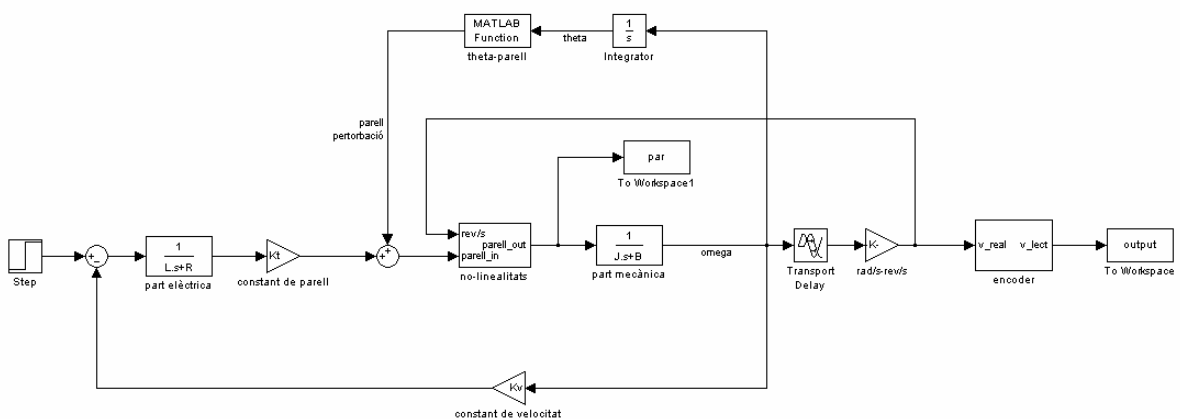


Figura 23: Diagrama de blocs de la planta amb el model del dispositiu magnètic (model Simulink).

La figura 23 mostra el diagrama de blocs del sistema en què s'ha inclòs el model del dispositiu magnètic<sup>14</sup>. L'ajust dels paràmetres del dispositiu s'ha realitzat de forma que la resposta simulada coincidís amb la resposta real. Així, els paràmetres utilitzats<sup>15</sup> són:

$\rho = [0, 3.2289];$       en radians  
 $\text{ofix} = [1, -1];$       adimensional

<sup>14</sup> La funció "parell.m" que calcula el parell de pertorbació sobre l'eix del motor a partir de la posició del propi eix i la configuració del dispositiu magnètic és a l'annex A. Cal incloure aquest arxiu en la mateixa carpeta que els models de Simulink per poder realitzar les simulacions.

<sup>15</sup> Els paràmetres estan guardats a l'arxiu "wsp\_planta\_pert.mat" a la carpeta \model\_físic\planta del CD adjunt. Aquest arxiu també inclou la resta de paràmetres de la planta.

$x = [0.0620, 0.0600];$  en metres  
 $rq = 0.8900;$  adimensional

També s'ha inclòs un bloc que simula l'encoder, amb la quantificació i el soroll<sup>16</sup>. La figura 24 presenta la resposta temporal simulada<sup>17</sup> solapada a la resposta real del sistema en llaç obert. Com es pot observar, el model ajusta prou bé la pertorbació real.

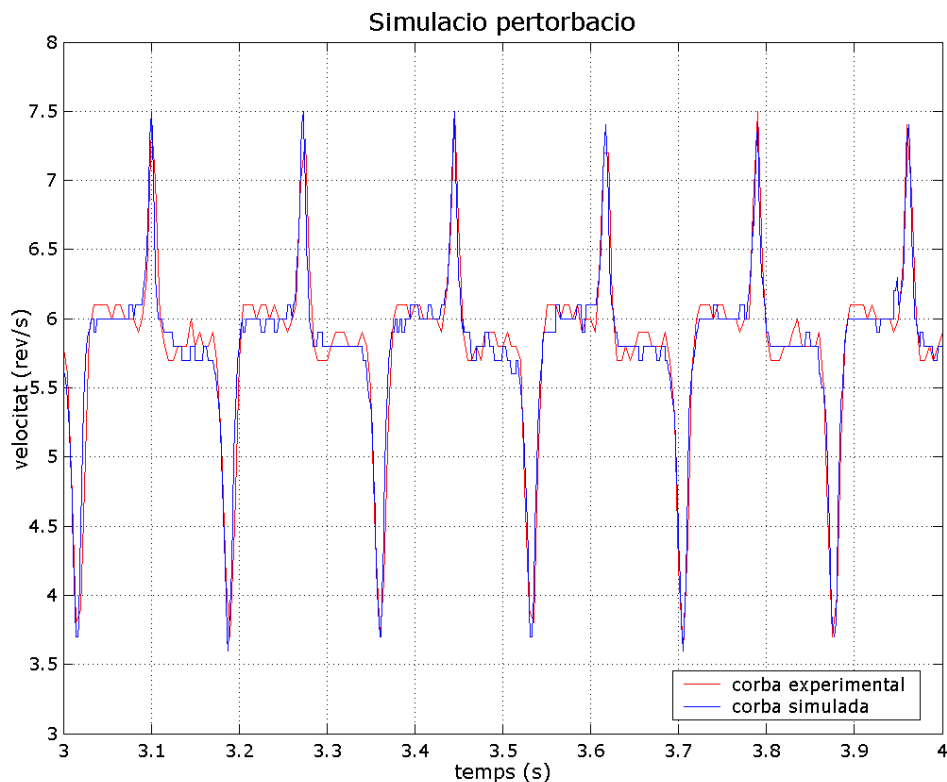


Figura 24: Resposta temporal simulada del model amb el dispositiu magnètic.

També és interessant analitzar l'espectre de la resposta del sistema amb la pertorbació. La figura 25 mostra l'espectre de potència obtingut a partir del senyal real a aproximadament 6rev/s en què s'ha eliminat la component de contínua. Com es pot observar, aquest presenta components freqüencials a 6Hz (aprox.), que és la component fonamental, i als seus harmònics. Aquí es pot veure clarament que la freqüència de les pertorbacions està acoplada amb la velocitat de gir de l'eix del motor.

<sup>16</sup> Si l'eix del motor dóna més d'una volta entre instants de mostratge, la lectura de la velocitat és inferior a la velocitat real de l'eix. La funció "module.m" simula aquest efecte i també cal incloure l'arxiu en la mateixa carpeta que els models de Simulink per poder realitzar les simulacions.

<sup>17</sup> L'arxiu "planta\_pert.mdl" permet realitzar simulacions amb el model obtingut. L'arxiu es troba a la carpeta \model\_físic\planta\.

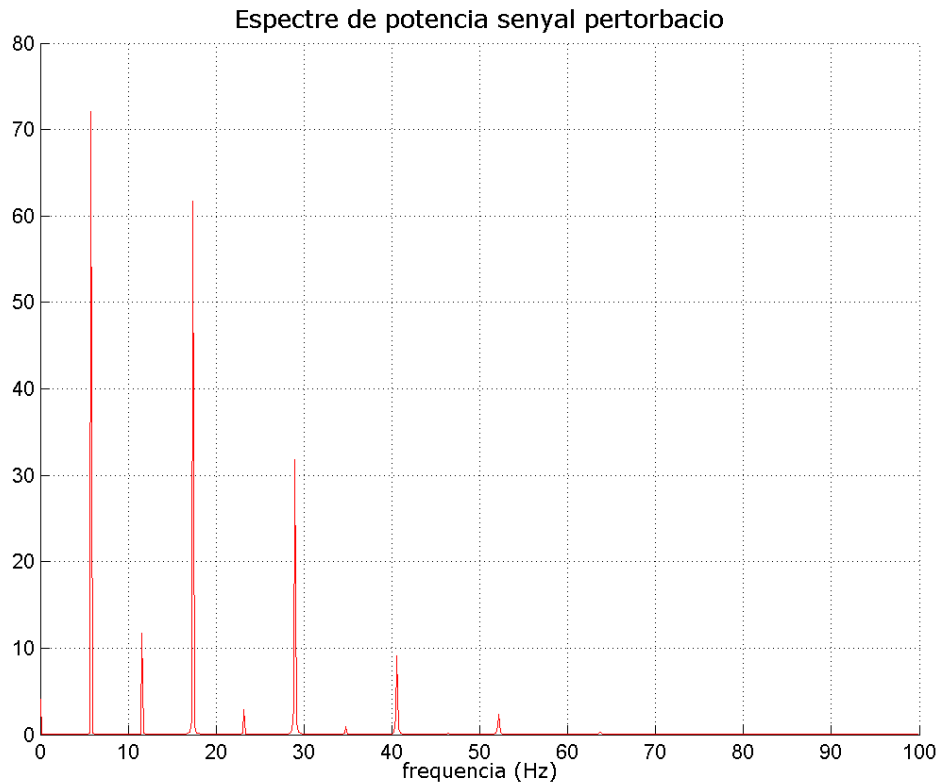


Figura 25: Espectre de potència del senyal de perturbació de la sortida de la planta enllaç obert (a partir de les dades experimentals).

### 4.3. Sistema de control

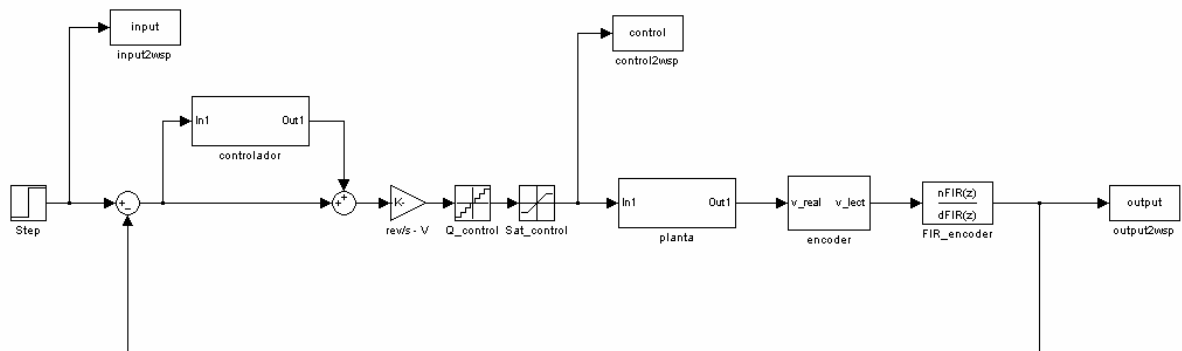


Figura 26: Diagrama de blocs de llaç tancat (model Simulink).

La figura 26 mostra el diagrama de blocs del sistema en llaç tancat. A part de tancar el llaç i incorporar el controlador en una estructura *plug-in*, s'ha afegit un bloc a la sortida de l'encoder que correspon a un filtre FIR passabaixos que pretén netejar el soroll de quantificació del senyal que surt de l'encoder. També s'ha afegit un bloc que satura el senyal de control i un altre que el quantifica.

Com que la planta és estable en llaç obert, ho és també en llaç tancat i, per tant, s'ha pres  $G_c(z)=1$  per criteris de simplicitat.

El controlador disposa d'un guany  $K_r$  ajustable que fixa la posició dels pols en llaç tancat i que ha de pertànyer a l'interval  $[0, 2]$  per complir les condicions d'estabilitat.

#### **4.4. Disseny del controlador**

Com ja s'ha comentat, el procediment de disseny del controlador està descrit amb detall a [4]. S'ha creat un *script*, "tf\_repe.m", i una funció, "gx.m", que retornen, entre d'altres coses, la funció de transferència del controlador repetitiu. De fet, només cal executar l'*script* "tf\_repe.m", ja que és aquest qui crida la funció "gx.m".

El disseny del controlador repetitiu depèn de la posició dels zeros de la planta en llaç tancat. Així, el mètode que cal seguir és un o és un altre en funció de si la planta és de fase mínima o és de fase no mínima, és a dir, si té o no té, respectivament, tots els zeros dins del cercle de radi unitat. Els únics zeros de la planta provenen del filtre FIR de la sortida de l'encoder i, per tant, segur que són de fase no mínima [5]. Només en el cas que no es posés aquest filtre FIR, la planta seria de fase mínima. Així doncs, l'*script* i la funció implementats<sup>18</sup> tenen en compte la posició dels zeros per a seguir un algorisme de disseny o un altre.

#### **4.5. Simulacions**

Es pretén controlar el sistema a una velocitat de 8rev/s, la qual cosa implica que el període de les pertorbacions és de 0.125s.

Tenint en compte que, com ja s'ha apuntat anteriorment, el període de mostratge del controlador ha d'anar sincronitzat amb el període de la pertorbació, es pren un període de mostratge de 5ms. Per tant, es tenen 25 mostres per període ( $N=25$ )<sup>19</sup>.

A més, a partir dels espectres anteriors, es pot observar que els harmònics del senyal de pertorbació amb una potència elevada es troben per sota de la freqüència de Nyquist, que en aquest cas és de 100Hz i, per tant, en principi es poden rebutjar.

---

<sup>18</sup> L'*script* "tf\_repe.m" i la funció "gx.m" són a l'annex A.

<sup>19</sup> Veure [4].



El disseny dels dos filtres, el de la sortida de l'encoder,  $FIR\_encod(z)$ , i el del controlador,  $H(z)$ , i el guany  $K_r$  que es dóna al controlador determinen com de bé es rebutja la pertorbació. Tanmateix, no hi ha un algorisme que permeti obtenir una resposta tancada òptima per a aquest problema. Tan sols es poden seguir uns pocs criteris:

- El filtre de sortida de l'encoder ha de ser un filtre passabaixos que netegi el soroll de quantificació. A partir de l'espectre del senyal de sortida en llaç obert i sense el dispositiu magnètic es podria intentar determinar la banda de pas que ha de tenir el filtre.
- El filtre del controlador serveix per donar-li robustesa i evitar, així, problemes deguts a la incertesa a l'hora de determinar els paràmetres de la planta. Estrictament, per tant, la seva banda passant hauria de coincidir amb el rang de freqüències en què es té ben identificada la planta. En el nostre cas, fins a uns 10Hz. Això implicaria, però, que no es podria rebutjar cap pertorbació per sobre d'aquests 10Hz i, per tant, en el nostre experiment a 8rev/s només es podria rebutjar la component fonamental. Per això, sovint, es dissenya aquest filtre perquè retalli només a altes freqüències, i més quan es tracta de plantes senzilles com la que és objecte d'estudi.
- Quan s'introdueix el filtre del controlador, es produeix un desplaçament dels pols del controlador, de manera que aquests s'allunyen de les freqüències que han de rebutjar. No es coneix cap estudi sobre el tema, però s'ha comprovat que, per minimitzar aquest efecte, interessa que aquest filtre sigui de fase nul·la i del menor ordre possible.

Així doncs, a partir d'aquests criteris i amb l'ajuda de la *toolbox* de MatLab de disseny i anàlisi de filtres (*FDATool*), s'han obtingut els filtres següents:

$$H(z) = 0.02z + .96 + .02z^{-1}$$

$$FIR\_encod(z) = \frac{0.04538z^5 + 0.1615z^4 + 0.3838z^3 + 0.3838z^2 + 0.1615z + 0.04538}{z^5}$$

$$(T_s=0.005s)$$

Les figures 27 i 28 mostren els diagrames de Bode dels dos filtres. De la funció de transferència es pot observar que el filtre  $H(z)$  és un filtre FIR no causal. Tanmateix, no hi ha cap problema a l'hora d'implementar el controlador, ja que

aquest filtre es troba en sèrie amb una cadena de retards purs. Al seu diagrama de Bode es pot veure que és de fase nul·la i que retalla molt poc (menys d'1dB a 100Hz). El filtre FIR de la sortida de l'encoder, en canvi, és un filtre FIR causal de fase lineal amb una banda de pas d'aproximadament 10Hz.

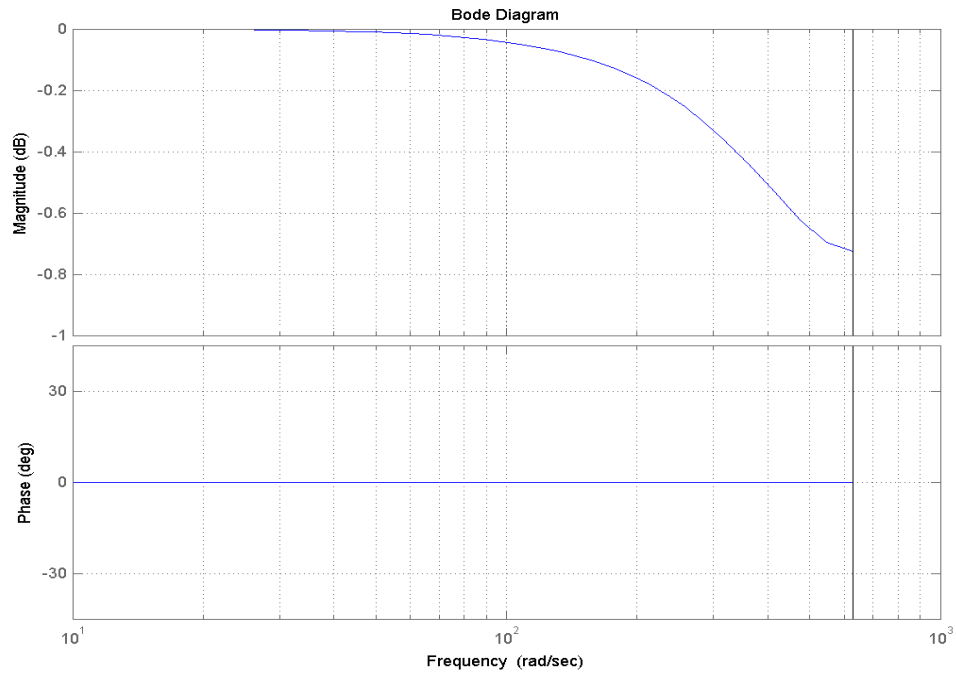


Figura 27: Diagrama de Bode del filtre del controlador  $H(z)$ .

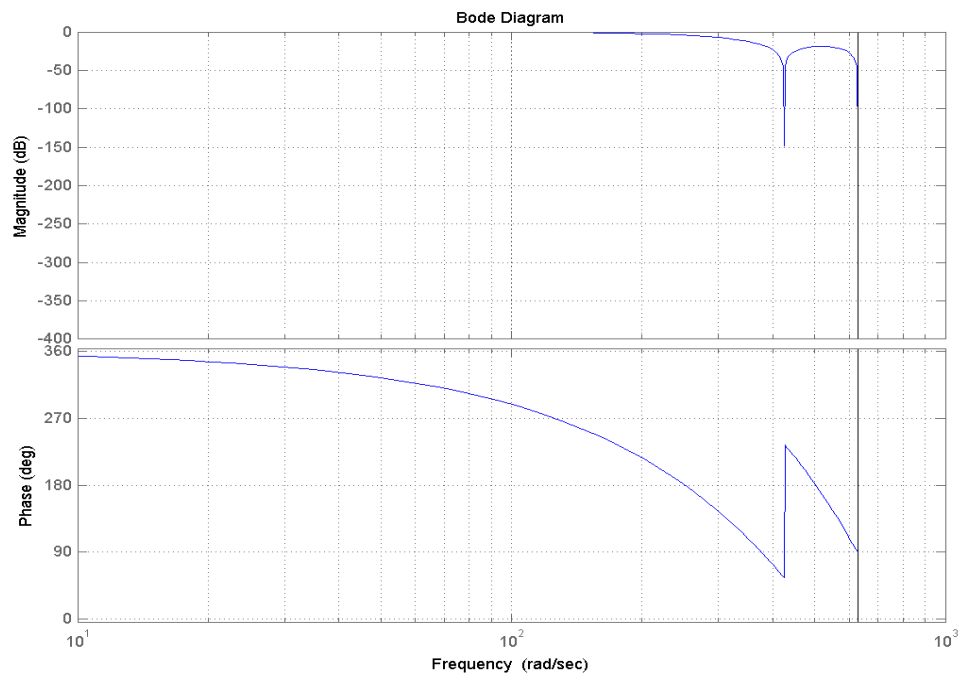


Figura 28: Diagrama de Bode del filtre de sortida de l'encoder  $FIR\_encod(z)$ .

La figura 29 mostra el mapa de pols i zeros de la planta en llaç tancat, la posició dels quals determina el procediment de disseny del controlador. Com es pot observar, apareixen 4 zeros de fase no mínima, tres sobre la circumferència de radi unitat i un sobre l'eix real positiu. Aquests zeros provenen del filtre FIR de sortida de l'encoder. A més, també apareix el que sembla una cancel·lació d'un pol i zero. Tanmateix, no és una cancel·lació física, ja que exactament no es troben sobre el mateix punt del pla Z. A efectes pràctics, però, es podria realitzar la cancel·lació i implementar el controlador sense tenir en compte aquest zero i aquest pol, amb la qual cosa reduiríem l'ordre del filtre.

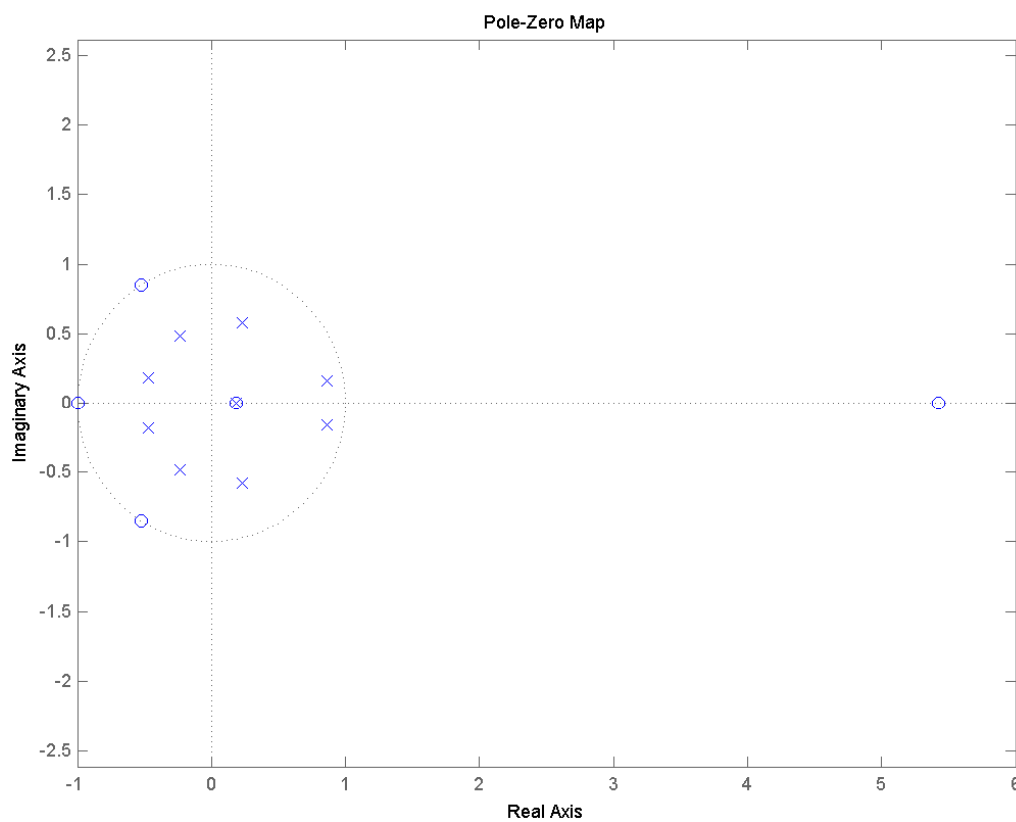


Figura 29: Mapa de pols i zeros de la planta en llaç tancat.

A més, també es pot observar que els pols de llaç tancat es troben dins la circumferència de radi unitat i, per tant, la planta és estable en llaç tancat. Això confirma el que s'havia dit a l'apartat 4.3. Si la planta presentés algun pol fora de la circumferència caldria dissenyar algun controlador  $G_c(z)$  que l'estabilitzés en llaç tancat.

Així, el controlador Grepe(z) que resulta del procediment de disseny adequat és:

```
>>Grepe=tf(nGrepe,dGrepe,Ts)
Transfer function:
0.03642 z^15 + 1.781 z^14 + 1.624 z^13 - 0.1978 z^12 - 1.742 z^11 - 1.398 z^10
+ 0.2935 z^9 + 0.107 z^8 + 0.2185 z^7 + 0.2643 z^6 + 0.1807 z^5 + 0.05911 z^4
- 0.006295 z^3 - 0.009247 z^2 + 0.001125 z + 2.741e-005
-----
z^31 - 0.1843 z^30 + 3.407e-014 z^29 + 1.193e-013 z^28 + 3.481e-013 z^27
+ 7.203e-013 z^26 + 9.634e-013 z^25 + 5.589e-013 z^24 - 9.908e-013 z^23
- 3.693e-012 z^22 - 6.983e-012 z^21 - 9.887e-012 z^20 - 1.11e-011 z^19
- 9.825e-012 z^18 - 6.904e-012 z^17 - 4.138e-012 z^16 - 2.413e-012 z^15
- 1.314e-012 z^14 - 4.431e-013 z^13 + 9.819e-014 z^12 + 2.24e-013 z^11
+ 1.371e-013 z^10 + 5.23e-014 z^9 + 1.259e-014 z^8 - 0.02 z^7 - 0.9563 z^6
+ 0.1569 z^5 + 0.003685 z^4

Sampling time: 0.005
```

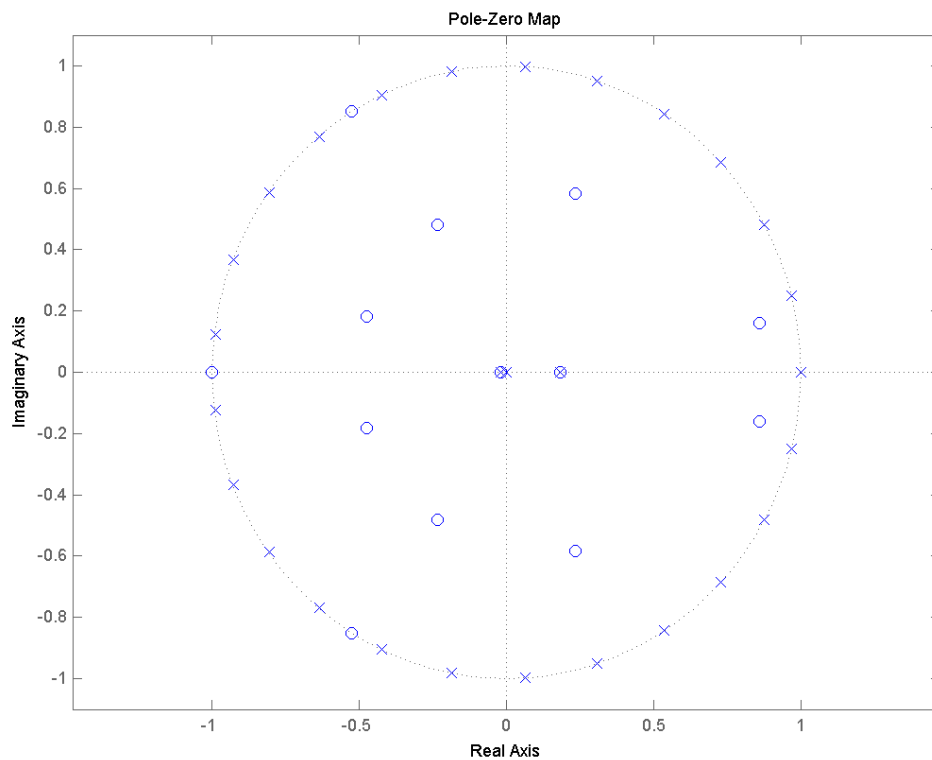


Figura 30: Mapa de pols i zeros del controlador dissenyat.

La figura 30 mostra el mapa de pols i zeros del controlador. Es pot observar una distribució uniforme de pols sobre la circumferència de radi unitat. Aquests pols són els que donen un guany infinit (o molt gran) a les freqüències d'interès per poder rebutjar la pertorbació amb un error nul en estat estacionari. A més, també

es poden observar dues possibles cancel·lacions. Una és la que es comentava abans, que ja s'ha analitzat i s'ha vist que no era una cancel·lació física. L'altra, però, sí és una cancel·lació física com es pot observar a partir de la informació proporcionada pel MatLab:

```
>> [z,p,k]=zpkdata(Grepe, 'v');
>> z

z =

-47.97915761656360
-0.02084238343640
 0.85910643416547 + 0.16121745202336i
 0.85910643416547 - 0.16121745202336i
-1.000000000000000
-0.52560698344237 + 0.85072751157854i
-0.52560698344237 - 0.85072751157854i
 0.23292809410126 + 0.58214917279850i
 0.23292809410126 - 0.58214917279850i
-0.47346375563181 + 0.18325589438203i
-0.47346375563181 - 0.18325589438203i
-0.23267170215655 + 0.48173846088757i
-0.23267170215655 - 0.48173846088757i
 0.18430185904448
 0.18426885678632

>> p

p =

1.000000000000000
0.96853444901506 + 0.24867735933176i
0.96853444901506 - 0.24867735933176i
0.87613291173050 + 0.48165796881568i
0.87613291173050 - 0.48165796881568i
0.72865118409856 + 0.68424834479935i
0.72865118409856 - 0.68424834479935i
0.53542613000315 + 0.84369465418041i
0.53542613000315 - 0.84369465418041i
0.30867240525550 + 0.94999030866882i
0.30867240525550 - 0.94999030866882i
0.06269715297643 + 0.99650238920810i
0.06269715297643 - 0.99650238920810i
-0.18701380120532 + 0.98037778429584i
-0.18701380120532 - 0.98037778429584i
-0.42477710608941 + 0.90270574969702i
-0.42477710608941 - 0.90270574969702i
-0.63569640713721 + 0.76843054685710i
-0.63569640713721 - 0.76843054685710i
-0.80658817005800 + 0.58602436193170i
-0.80658817005800 - 0.58602436193170i
-0.98882470675648 + 0.12491833769131i
-0.98882470675648 - 0.12491833769131i
-0.92679285011458 + 0.36694544652139i
-0.92679285011458 - 0.36694544652139i
-0.02084238343640
 0
 0
 0
 0
 0.18426885678753
```

Per tant, convé realitzar la cancel·lació d'aquest pol amb el zero corresponent. La figura 31 mostra el mapa de pols i zeros del nou controlador, en què ja no apareix la cancel·lació anterior.

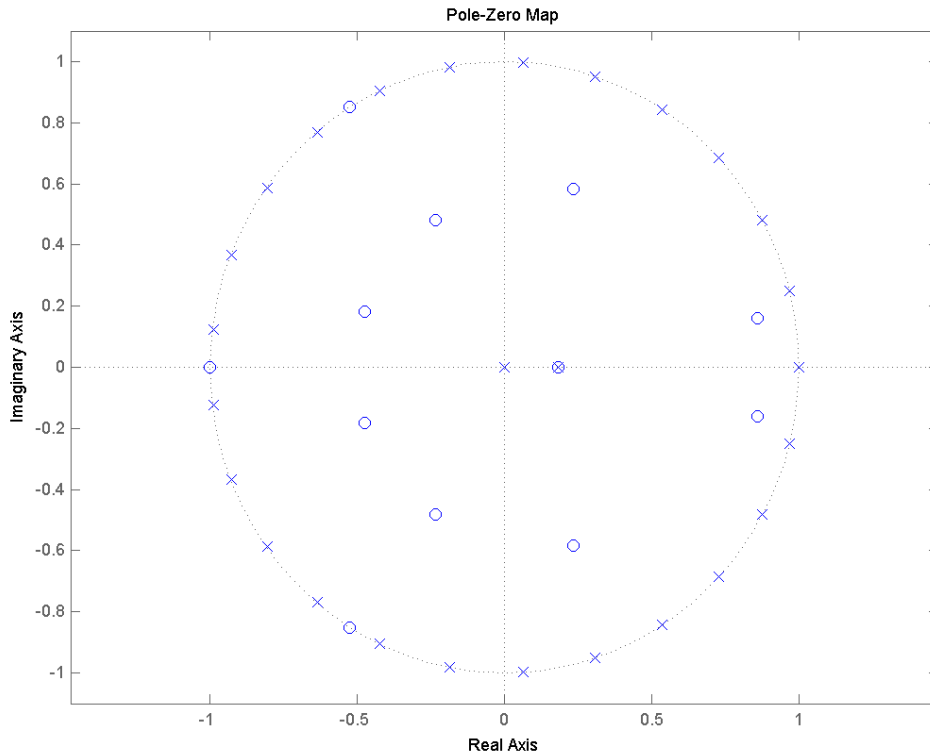


Figura 31: Mapa de pols i zeros del nou controlador.

I la nova funció de transferència és:

```
>> Grepe2
Transfer function:
  0.03642 z^14 + 1.78 z^13 + 1.587 z^12 - 0.2308 z^11 - 1.737 z^10 - 1.361 z^9
    + 0.3219 z^8 + 0.1002 z^7 + 0.2164 z^6 + 0.2598 z^5 + 0.1753 z^4
      + 0.05545 z^3 - 0.007451 z^2 - 0.009092 z + 0.001315
-----
z^30 - 0.2051 z^29 + 0.004275 z^28 - 8.91e-005 z^27 + 1.857e-006 z^26
    - 3.871e-008 z^25 + 8.078e-010 z^24 - 1.967e-011 z^23 + 5.339e-012 z^22
    - 5.871e-012 z^21 + 3.723e-012 z^20 + 1.78e-012 z^19 - 9.061e-012 z^18
    + 1.525e-011 z^17 - 1.783e-011 z^16 + 1.611e-011 z^15 - 1.153e-011 z^14
    + 6.56e-012 z^13 - 2.964e-012 z^12 + 1.047e-012 z^11 - 2.837e-013 z^10
    + 3.342e-014 z^9 + 5.884e-015 z^8 + 4.996e-015 z^7 - 0.02 z^6 - 0.9559 z^5
      + 0.1768 z^4

Sampling time: 0.005
```

De la funció de transferència es pot observar que la majoria de coeficients del denominador són molt petits comparats amb els coeficients dels primers i darrers termes. Es pot provar de forçar que aquests coeficients siguin nuls, la qual cosa faria més senzilla la implementació real del controlador. La figura 32 mostra el mapa de pols i zeros per als dos controladors, on es pot observar que no hi ha

diferències significatives. La funció de transferència per a aquest nou controlador es presenta a continuació:

```
>> Grepe3
Transfer function:
0.03642 z^14 + 1.78 z^13 + 1.587 z^12 - 0.2308 z^11 - 1.737 z^10 - 1.361 z^9
+ 0.3219 z^8 + 0.1002 z^7 + 0.2164 z^6 + 0.2598 z^5 + 0.1753 z^4
+ 0.05545 z^3 - 0.007451 z^2 - 0.009092 z + 0.001315
-----
z^30 - 0.2051 z^29 + 0.004275 z^28 - 0.02 z^6 - 0.9559 z^5 + 0.1768 z^4
Sampling time: 0.005
```

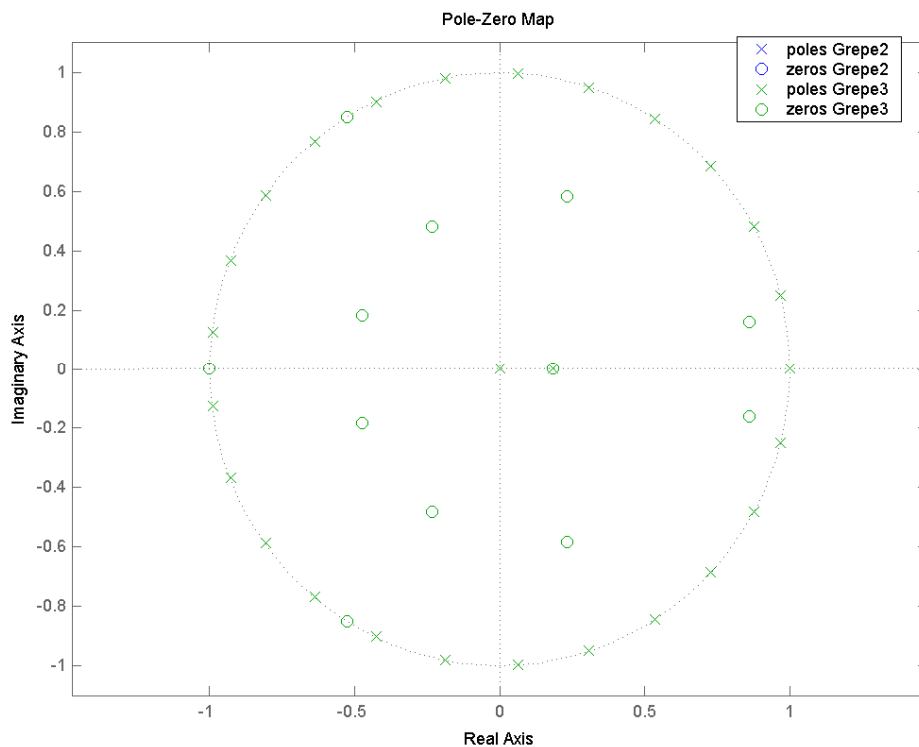


Figura 32: Mapa de pols i zeros del tercer controlador.

La figura 33 mostra el Diagrama de Bode d'aquest tercer i definitiu controlador. Es poden observar els pics de guany situats a 8Hz ( $\sim 50.26\text{rad/s}$ ) i als seus harmònics. També es pot observar l'atenuació d'aquests pics a altes freqüències provocada pel filtre  $H(z)$  que dona robustesa al controlador.

Així doncs, ja s'ha realitzat el disseny del controlador que, teòricament, ha de rebutjar la pertorbació introduïda pel dispositiu magnètic. L'únic paràmetre que queda per ajustar és el guany  $K_r$  del controlador, que dependrà de les especificacions que hagi de complir l'aplicació.

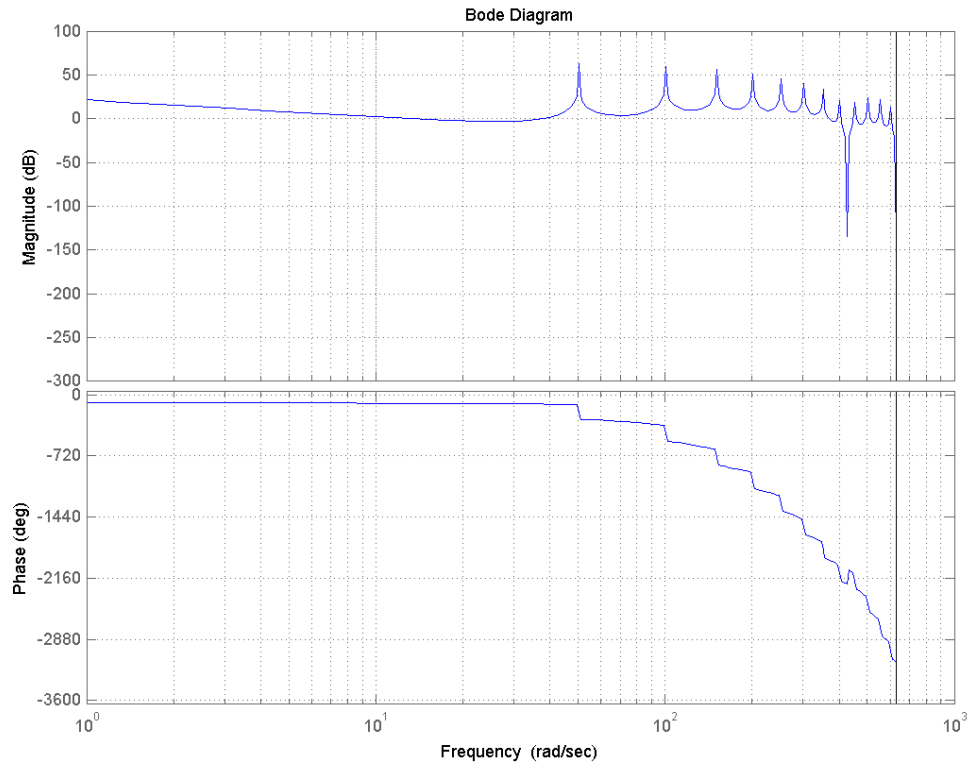


Figura 33: Diagrama de Bode del controlador repetitiu dissenyat.

Les figures 34 a 37 mostren la resposta temporal i l'acció de control del sistema per a diferents valors de  $K_r$ . Com es pot observar, la velocitat de sortida es manté aproximadament constant al voltant de les 8rev/s, de manera que es rebutgen, en gran mesura, les pertorbacions creades pel dispositiu magnètic. També es pot observar que la resposta del sistema és més ràpida en augmentar el guany del controlador. Però cal tenir present que, augmentant aquest guany, s'està empijorant l'estabilitat del sistema.

Els efectes del controlador també es poden observar a partir dels espectres de potència del senyal de sortida. Les figures 38 i 39 permeten comparar els espectres de potència del senyal de pertorbació (sense la component de contínua) i del senyal de sortida en llaç tancat per a un valor de  $K_r$  de 0.8. Com es pot observar, s'han rebutjat completament les components de freqüència corresponents als quatre primers harmònics del senyal de pertorbació (dins de la banda de pas dels filtres dissenyats) i s'han atenuat significativament els harmònics superiors.



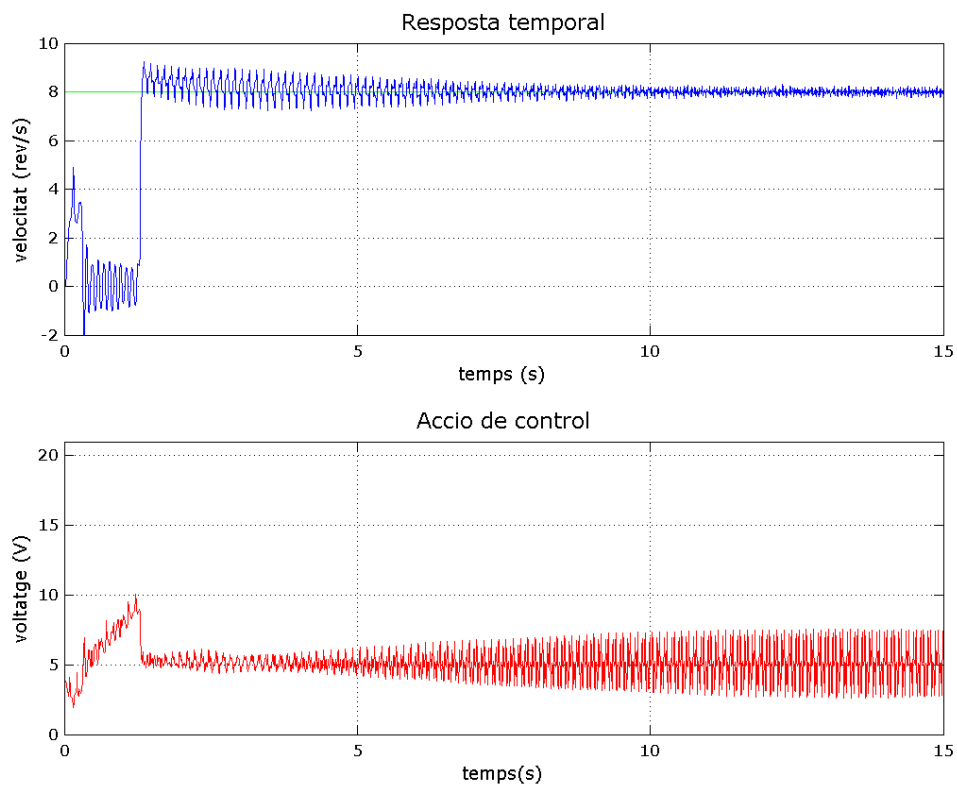


Figura 34: Simulació per a  $K_r = 0.1$ .

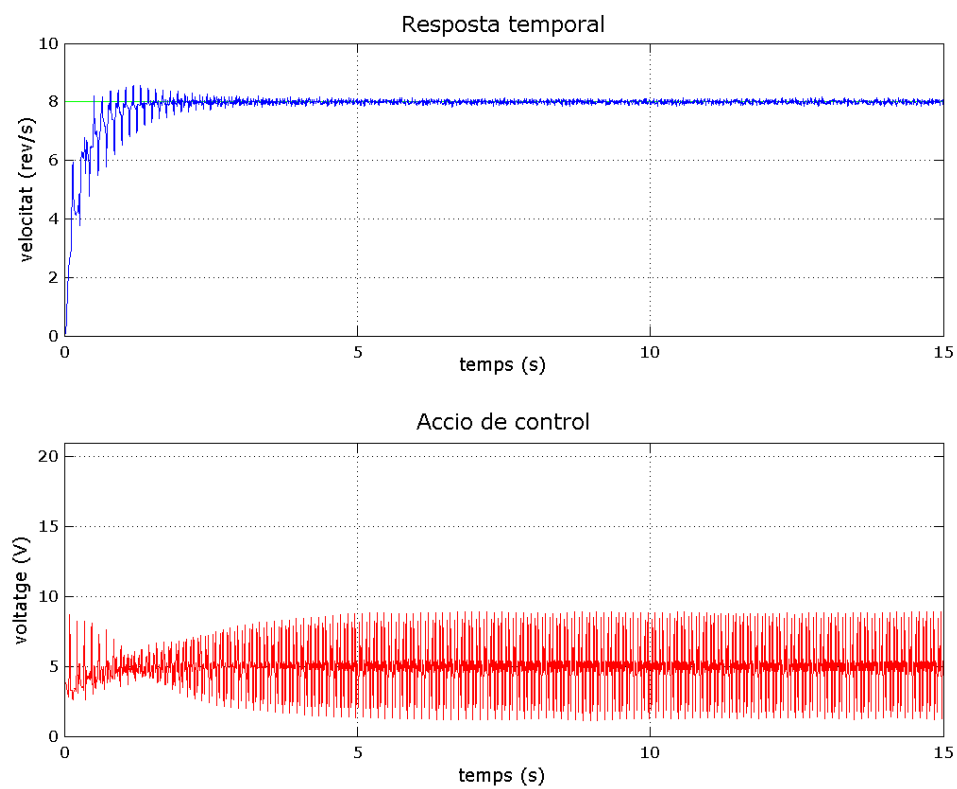


Figura 35: Simulació per a  $K_r = 0.4$ .

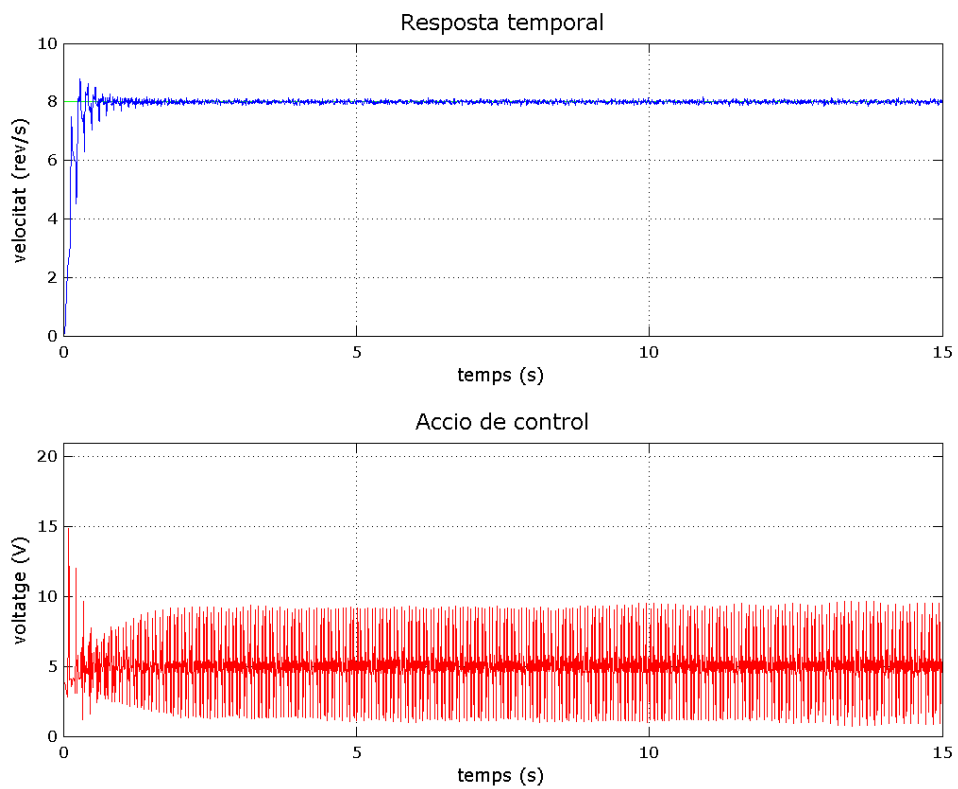


Figura 36: Simulació per a  $K_r = 0.8$ .

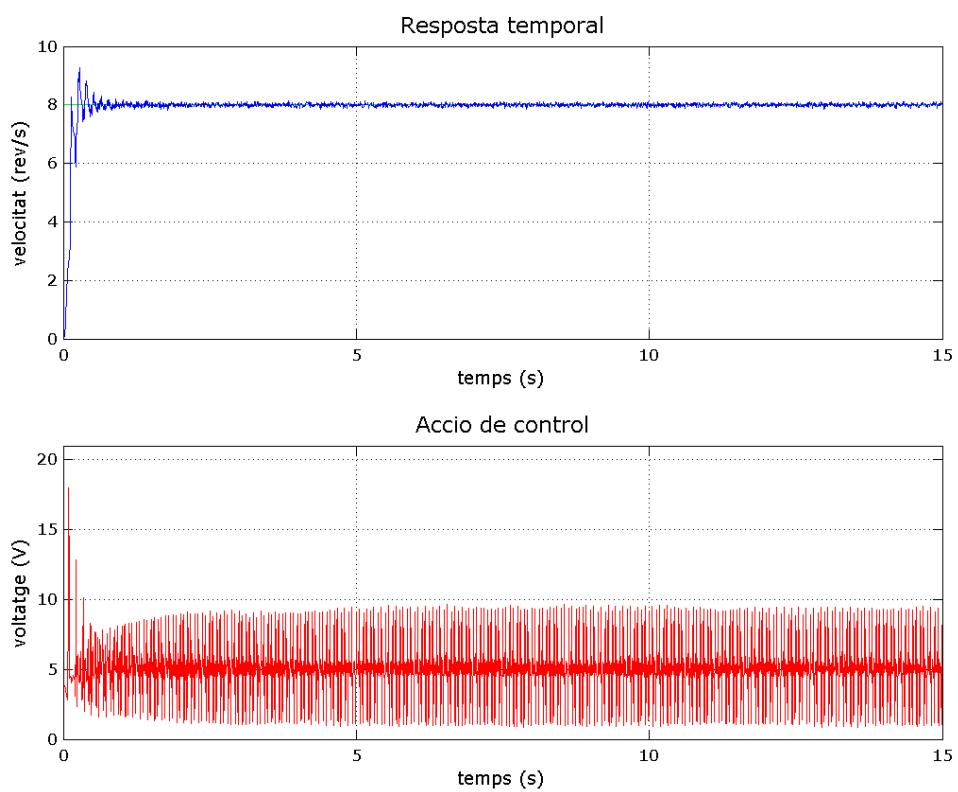


Figura 37: Simulació per a  $K_r = 1$ .

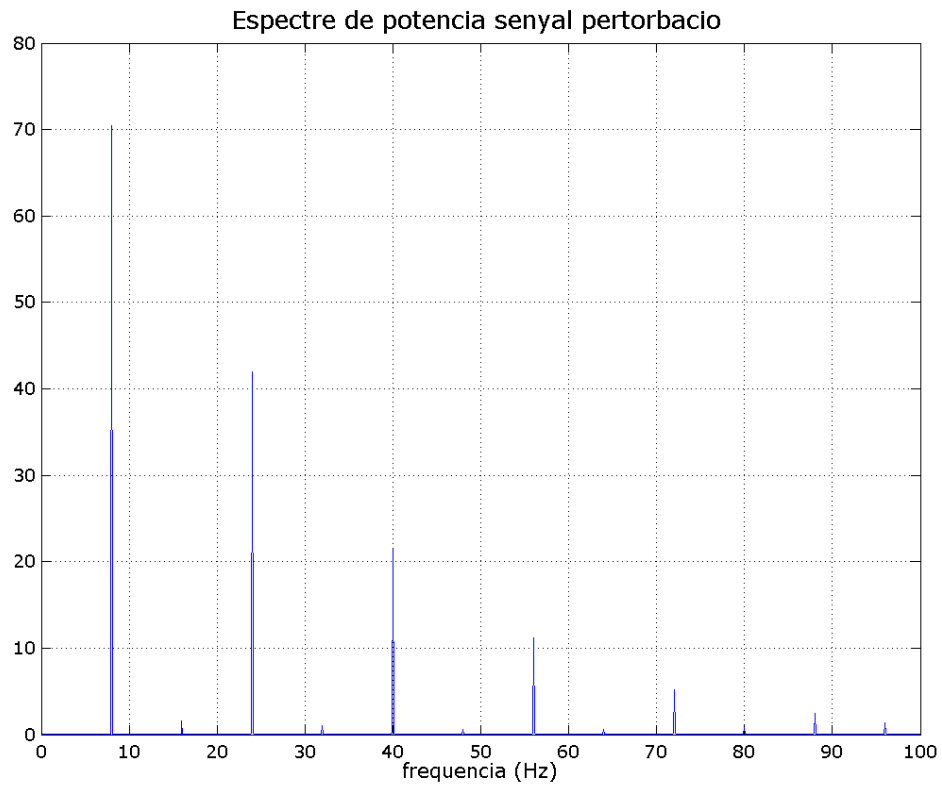


Figura 38: Espectre de potència del senyal de pertorbació de la sortida de la planta en llac obert (a partir de dades simulades).

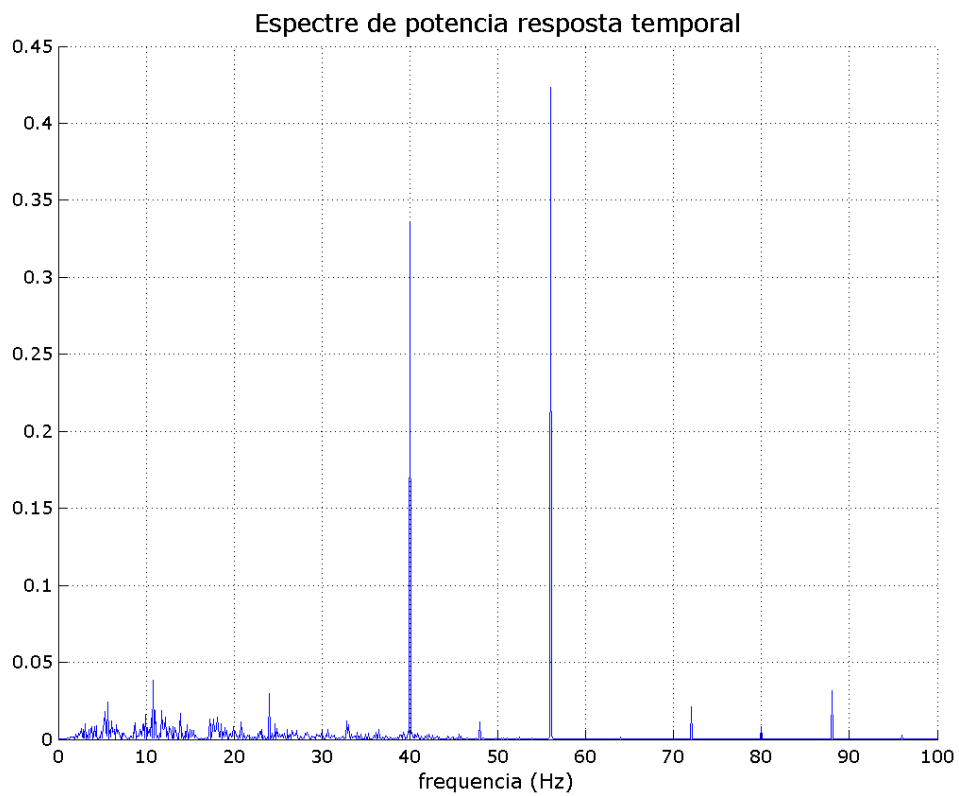


Figura 39: Espectre de potència del senyal de sortida de la planta en llac tancat per a  $K_r = 0.8$  (a partir de dades simulades).

## **CONCLUSIONS**

En aquest treball s'ha presentat i analitzat el comportament d'una planta didàctica especialment dissenyada per a l'experimentació d'algorismes de control per al seguiment o rebuig de senyals periòdics. S'han obtingut tres models que en descriuen el comportament seguint tres metodologies diferents. En aquest sentit, s'han utilitzat algunes de les eines de què disposa MatLab per a la identificació de sistemes, que s'han mostrat molt útils i potents.

El treball també inclou un exemple en què es prova la validesa del control repetitiu per fer front a les pertorbacions existents al sistema. Aquest exemple ha permès, també, validar el model lineal obtingut amb MatLab.

Per a la realització de simulacions, tant per la part d'identificació com per la de control repetitiu, s'ha comprovat la versatilitat del Simulink i la riquesa de les seves llibreries, que ha facilitat de forma notable el disseny de les no-linealitats del sistema.

Així, s'ha vist que MatLab ofereix unes possibilitats molt àmplies a la gent que treballa en l'àmbit de l'Automàtica i que pot ser, en la majoria de projectes, una eina molt útil de suport en l'anàlisi i disseny de sistemes, tant en llaç obert com en llaç tancat.

## **BIBLIOGRAFIA**

- [1] E. Xargay, "Modelat d'un sistema amb pertorbacions periòdiques i implementació d'un controlador repetitiu de freqüència variable", Miniprojecte de Tecnologia de Control, Juny 2004.
- [2] Autor desconegut, "Procedure for identifying permanent magnet DC motors", Virginia Tech Mechatronics.  
<<http://www.mechatronics.me.vt.edu/book/Section3/motormodelling.html>>
- [3] B. Francis and W. Wonham, "Internal model principle in control theory", *Automatica*, vol. 12, pp. 457-465, 1976.
- [4] R. Costa, J. Nebot i R. Griñó, "Digital repetitive controller for an educational laboratory plant", IOC, December 2003.
- [5] J. G. Proakis and D. G. Manolakis, *Tratamiento digital de señales*, 3ª edición ed. Prentice Hall, 1998.

## **Annex A**

### **CODIS MATLAB**

En aquest annex es presenten, comentats, els codis dels principals *scripts* i funcions utilitzats en l'elaboració del treball i als quals s'ha anat fent referència durant el seu desenvolupament. Els arxius corresponents es poden trobar al CD adjunt.

### \*\*\* plot\_c\_estat.m \*\*\*

```
%Aquest script llegeix les dades experimentals obtingudes per a la
%determinacio de la corba estatica de la planta dels arxius c_estat_up.out
%i c_estat_down.out, calcula la velocitat mitjana de sortida per a cada
%nivell de tensio i dibuixa la corba estatica.

last_value=40000;
first_value=301;

%UP
%es llegeixen les dades de l'arxiu .out
[voltage,velo]=textread('c_estat_up.out', '%*f %*d %*f %f %f', last_value);

pos_i=first_value;
i=first_value+1;
j=1;

%per a cada valor de tensio es calcula la velocitat de sortida mitjana a
%partir de les dades en estat estacionari
while i<last_value

    while (voltage(i)==voltage(i-1) & i<last_value)
        i=i+1;
    end

    if i==last_value
        pos_f=i;
    else
        pos_f=i-1;
    end

    VOLTAGE(j)=voltage(pos_i);
    VELO(j)=mean(velo(pos_i:pos_f));

    pos_i=i+300;
    i=i+301;
    j=j+1;
end

%es dibuixen els resultats
plot(VOLTAGE,VELO,'r. ');
hold on;

%DOWN

[voltage,velo]=textread('c_estat_down.out', '%*f %*d %*f %f %f', last_value);

pos_i=first_value;
i=first_value+1;
j=1;

while i<last_value

    while (voltage(i)==voltage(i-1) & i<last_value)
        i=i+1;
    end

    if i==last_value
        pos_f=i;
    else
        pos_f=i-1;
    end

    VOLTAGE(j)=voltage(pos_i);
    VELO(j)=mean(velo(pos_i:pos_f));

    pos_i=i+300;
    i=i+301;
    j=j+1;
end

plot(VOLTAGE,VELO,'ro ');
grid;
```

\*\*\* sys\_gain.m \*\*\*

```
function[gain]=sys_gain()

%Aquesta funcio retorna el guany canonic de la planta per a tensions
%d'entrada de la planta positives.
%El calcul es realitza a partir de les dades dels arxius c_estat_up.out i
%c_estat_down.out.

last_value=40000;
first_value=301;

%UP

[voltage,velo]=textread('c_estat_up.out', '%*f %*d %*f %f %f', last_value);

pos_i=first_value;
i=first_value+1;
j=1;

while i<last_value

    while (voltage(i)==voltage(i-1) & i<last_value)
        i=i+1;
    end

    if i==last_value
        pos_f=i;
    else
        pos_f=i-1;
    end

    data_up.voltage(j)=voltage(pos_i);
    data_up.velo(j)=mean(velo(pos_i:pos_f));

    pos_i=i+300;
    i=i+301;
    j=j+1;
end

%DOWN

[voltage,velo]=textread('c_estat_down.out', '%*f %*d %*f %f %f', last_value);

pos_i=first_value;
i=first_value+1;
j=1;

while i<last_value

    while (voltage(i)==voltage(i-1) & i<last_value)
        i=i+1;
    end

    if i==last_value
        pos_f=i;
    else
        pos_f=i-1;
    end

    data_down.voltage(j)=voltage(pos_i);
    data_down.velo(j)=mean(velo(pos_i:pos_f));

    pos_i=i+300;
    i=i+301;
    j=j+1;
end

%REGRESSION UP
%s'ajusta una recta a les dades per minims quadrats
data_up.voltage=data_up.voltage';
data_up.velo=data_up.velo';
```



```

pos_i=find(data_up.voltage==6);
pos_f=find(data_up.voltage==20);

VOLTAGE=[ones(size(data_up.voltage(pos_i:pos_f))) data_up.voltage(pos_i:pos_f)];

a1= VOLTAGE\data_up.velo(pos_i:pos_f);

%REGRESSION DOWN
%s'ajusta una recta a les dades per minims quadrats
data_down.voltage=data_down.voltage';
data_down.velo=data_down.velo';

pos_f=find(data_down.voltage==6);
pos_i=find(data_down.voltage==20);

VOLTAGE=[ones(size(data_down.voltage(pos_i:pos_f))) data_down.voltage(pos_i:pos_f)];

a2= VOLTAGE\data_down.velo(pos_i:pos_f);

%REGRESSION
%s calcula el valor mig dels coeficients
a=(a1+a2)/2;
%s retorna el pendent de la recta de regressio
gain=a(2);

```

### \*\*\* plot\_resposta.m \*\*\*

```
%Aquest script llegeix les dades experimentals de la resposta temporal de la planta
%de l'arxiu resposta.out i dibuixa la resposta del sistema pel primer grao.

STEPS=3;

SAMPLES_PER_STEP=400;
SAMPLES=STEPS*SAMPLES_PER_STEP*2;
FIRST_SAMPLE=201;

%es llegeixen les dades de l'arxiu .out
[time, voltage, velo]=textread('resposta.out','%f %*d %*f %f %f', SAMPLES);

%es dibuixa la resposta pel primer grao
plot(time(FIRST_SAMPLE:FIRST_SAMPLE+2*SAMPLES_PER_STEP),
      velo(FIRST_SAMPLE:FIRST_SAMPLE+2*SAMPLES_PER_STEP), 'b. ');
axis([1 5 4 18]);
grid;
```

### \*\*\* sys\_tau.m \*\*\*

```
function[tau_mean]=sys_tau()

%Aquesta funcio retorna la constant de temps dominant del sistema.
%El calcul es realitza a partir de les dades de l'arxiu resposta.out.

%Nombre de graons a partir dels quals es calcula la constant de temps
%En el cas de l'arxiu resposta.out, el maxim es 3
STEPS=3;
%caracteristiques de l'experiment
SAMPLES_PER_STEP=400;
SAMPLES=STEPS*SAMPLES_PER_STEP*2;
FIRST_SAMPLE=201;

[time, voltage, velo]=textread('resposta.out','%f %d %f %f %f', SAMPLES);

%es guarden les dades de manera que es faciliti el tractament posterior
for j=1:1:STEPS

    samples.time(j,:)=time(FIRST_SAMPLE+(j-1)*2*SAMPLES_PER_STEP:j*2*SAMPLES_PER_STEP);
    samples.voltage(j,:)=voltage(FIRST_SAMPLE+(j-1)*2*SAMPLES_PER_STEP:j*2*SAMPLES_PER_STEP);
    samples.velo(j,:)=velo(FIRST_SAMPLE+(j-1)*2*SAMPLES_PER_STEP:j*2*SAMPLES_PER_STEP);
end

%per a cada grao es calcula la constant de temps
for j=1:1:STEPS

    velo_ini=mean(samples.velo(j,101:200));
    velo_final=mean(samples.velo(j,501:600));
    step=velo_final-velo_ini;
    value=velo_ini+.632*step;
    i=1;
    while samples.velo(j,i)<value
        i=i+1;
    end
    time63=interp1(samples.velo(j,i-1:i),samples.time(j,i-1:i),value,'linear');
    k=201;
    while samples.velo(j,k)<=6
        k=k+1;
    end
    time_ini=samples.time(j,k-1);
    tau(j)=time63-time_ini;
end

%es retorna la mitjana de les constant de temps calculades
tau_mean=mean(tau);
```

\*\*\* espectre\_prbs.m \*\*\*

```
function[]=espectre_prbs(time, signal);

%Aquesta funcio dibuixa l'espectre d'un senyal signal d'entrada.
%Els valors del vector time d'entrada presenten increments constants.
%S'utilitzen totes les dades del senyal signal.

%es calcula l'increment de temps Ts
Ts=time(2)-time(1);

%es calcula la FFT del senyal signal i en guardem la magnitud
N=size(signal,1);
SIGNAL=fft(signal,N);
Psignal=SIGNAL.*conj(SIGNAL);
Esignal=sqrt(Psignal)/N;
%es calculen les frequencies
freqs=(1/Ts)*(0:N/2-1)/N;

%es dibuixa l'espectre en eixos logaritmics
loglog(freqs,Esignal(1:N/2),'b');
```

### \*\*\* prbs\_adapt.m \*\*\*

```
function[iden, valid, ok]=prbs_adapt()

%Aquesta funcio retorna les dades experimentals d'identificacio i validacio
%adaptades als nivells adequats per a realitzar l'identificacio del sistema.
%Els arxius que contenen les dades necessaries son els arxius prbs_iden.out
%i prbs_valid.out.

%caracteristiques de l'experiment
PRBS_VALUES=4048;
DELAY_SAMPLES=400;

SAMPLES=DELAY_SAMPLES+PRBS_VALUES;

pos_i=DELAY_SAMPLES+1;
pos_f=SAMPLES;

%DADES IDENTIFICACIO
%es llegeixen les dades de l'arxiu .out
[time, voltage, velo]=textread('prbs_iden.out','%f %d %f %f %f %f',SAMPLES);
time=time(pos_i:pos_f);
voltage=voltage(pos_i:pos_f);
velo=velo(pos_i:pos_f);

%es comproven que s'hagin llegit totes les dades
iden_ok=0;
final=textread('prbs_iden.out','%f %d %f %f %f %f',SAMPLES+1);
if final(pos_f)==0 & final(pos_f+1)==1
    iden_ok=1;
end
clear final;

%adaptacio dades
%offset i escalat
voltage_max=max(voltage);
voltage_min=min(voltage);
voltage_mean=(voltage_max+voltage_min)/2;
voltage_amp=voltage_max-voltage_min;
voltage_adap=(voltage-voltage_mean)/(voltage_amp/2);
%
velo_max=max(velo);
velo_min=min(velo);
velo_mean=(velo_max+velo_min)/2;
velo_amp=velo_max-velo_min;
velo_adap=(velo-velo_mean)/(velo_amp/2);
%data struct
iden.time=time;
iden.voltage=voltage_adap;
iden.velo=velo_adap;

%DADES VALIDACIO
[time, voltage, velo]=textread('prbs_valid.out','%f %d %f %f %f %f',SAMPLES);
time=time(pos_i:pos_f);
voltage=voltage(pos_i:pos_f);
velo=velo(pos_i:pos_f);

valid_ok=0;
final=textread('prbs_valid.out','%f %d %f %f %f %f',SAMPLES+1);
if final(pos_f)==0 & final(pos_f+1)==1
    valid_ok=1;
end
clear final;

ok=iden_ok*valid_ok;

%
voltage_max=max(voltage);
voltage_min=min(voltage);
voltage_mean=(voltage_max+voltage_min)/2;
voltage_amp=voltage_max-voltage_min;
voltage_adap=(voltage-voltage_mean)/(voltage_amp/2);
%
velo_max=max(velo);
```

```
velo_min=min(velo);  
velo_mean=(velo_max+velo_min)/2;  
velo_amp=velo_max-velo_min;  
velo_adap=(velo-velo_mean)/(velo_amp/2);  
%data struct  
valid.time=time;  
valid.voltage=voltage_adap;  
valid.velo=velo_adap;
```

### \*\*\* parell.m \*\*\*

```
function[parell]=parell(theta, rho, ofix, x, rq)

%Aquesta funcio retorna el parell creat del dispositiu magnetic en funcio
%de la posicio theta de l'eix del motor.
%
%rho, ofix i x permeten ajustar la configuracio del dispositiu magnetic.
%rho, ofix i x son vectors de dimensions [1 ... N] on N es el nombre d'imants
%fixos del dispositiu.
%rho es l'angle on es col.loca l'imant fix, x la distancia a l'origen i
%ofix es "l'orientacio" dels imants fixos. {1} indica que l'imant fix te el pol
%nord mirant enfora, mentre que {-1} indica que el te mirant endins.
%rq es un escalar entre 0 i 1, i afecta la intensitat del pol magnetic de
%l'imant mobil 2.

%parametres del model: radi, permeabilitat, moment magnetic, gruix imants,
%intensitats dipols
r=0.05;
mu=4*pi*10^(-7);
m=23;
e=0.008;
qm1=1;
qm2=qm1*rq;

%calcul numero imants fixos N
N=size(rho,2);

%PARELL IMANT MOBIL 1

%calcul camp magnetic
for j=1:1:N

    %calcul coordenades alpha i d per a l'imant fix j
    alpha=atan(r*sin(theta-rho(j))/(r*cos(theta-rho(j))-x(j)));
    alpha=pi+alpha;
    d=(r*cos(theta-rho(j))-x(j))/cos(alpha);

    %calcul components del camp en refl creat per a l'imant fix j
    Bd=ofix(j)*(2*mu*m/(4*pi))*(cos(alpha)/(d^3));
    Balpha=ofix(j)*(mu*m/(4*pi))*(sin(alpha)/(d^3));

    %calcul components del camp en ref R creat per l'imant fix j
    Br(j)=Bd*cos(alpha-(theta-rho(j)))-Balpha*sin(alpha-(theta-rho(j)));
    Btheta(j)=Bd*sin(alpha-(theta-rho(j)))+Balpha*cos(alpha-(theta-rho(j)));
end

%calcul camp total (nomes component tangencial)
B_theta=0;
for j=1:1:N
    B_theta=B_theta+Btheta(j);
end

%calcul parell sobre imant mobil
parell_1=qm1*B_theta*e;

%PARELL IMANT MOBIL 2

%calcul camp magnetic
thetaux=theta+pi;
for j=1:1:N

    %calcul coordenades alpha i d per a l'imant fix j
    alpha=atan(r*sin(thetaux-rho(j))/(r*cos(thetaux-rho(j))-x(j)));
    alpha=pi+alpha;
    d=(r*cos(thetaux-rho(j))-x(j))/cos(alpha);

    %calcul components del camp en refl creat per a l'imant fix j
    Bd=ofix(j)*(2*mu*m/(4*pi))*(cos(alpha)/(d^3));
    Balpha=ofix(j)*(mu*m/(4*pi))*(sin(alpha)/(d^3));

    %calcul components del camp en ref R creat per l'imant fix j
    Br(j)=Bd*cos(alpha-(thetaux-rho(j)))-Balpha*sin(alpha-(thetaux-rho(j)));
    Btheta(j)=Bd*sin(alpha-(thetaux-rho(j)))+Balpha*cos(alpha-(thetaux-rho(j)));
end
```

```

%calcul camp total (nomes tangencial)
B_theta=0;
for j=1:1:N
    B_theta=B_theta+Btheta(j);
end

%calcul parell sobre imant mobil
parell_2=-qm2*B_theta*e;

%calcul parell total
parell=parell_1+parell_2;

```



### \*\*\* tf\_repe.m \*\*\*

```
%Aquest script calcula la funcio de transferencia del controlador repetitiu
%a partir dels numeradors dels dos filtres FIR necessaris.
%També genera les variables necessaries per a realitzar simulacions amb el
%model control.mdl de Simulink.
```

```
%periode de mostratge del controlador
Ts=.005;
%nombbre de mostres per periode de pertorbacio
N=25;
```

```
% MODEL PLANTA
```

```
%motor
nGp_D=.0900;
dGp_D=[1 -.9561 0 0 0];
Gp_D=tf(nGp_D,dGp_D,Ts);
%filtre FIR
nFIR=num_FIR_out;
gain=1/sum(nFIR);
nFIR=gain*nFIR;
dFIR=zeros(1,size(nFIR,2));
dFIR(1)=1;
G_FIR=tf(nFIR,dFIR,Ts);
%sys motor+FIR
sys_D=Gp_D*G_FIR;
```

```
% CONTROLADOR REPETITIU
```

```
%Gc
Gc=1;
%Gx
Go=feedback(sys_D,1);
Gx=gx(Go,Ts);
%retards
nGret=1;
dGret=[1,zeros(1,N)];
G_ret=tf(nGret,dGret,Ts);
%filtre H
nH=num_FIR_cont;
gain=1/sum(nH);
nH=gain*nH;
dH=[1,zeros(1,floor(size(nH,2)/2))];
G_H=tf(nH,dH,Ts);
%funcio de transferencia del controlador
Gr=feedback(G_H*G_ret,1,1);
%
Grepe=Gr*Gx;
%guany Kr del controlador
Kr=.5;
%
[nGrepe,dGrepe]=tfdata(Grepe,'v');
```

\*\*\* gx.m \*\*\*

```
function[Gx]=gx(Go,Ts)

%Aquesta funcio retorna la funcio de transferencia Gx pel disseny del
%controlador repetitiu.

%definim una tolerancia per problemes de quantificacio
TOLER=1e-10;

%zeros de la planta en llaç tancat
Z=zero(Go);

%el procediment de disseny es diferent si la planta es de fase minima o si
%es de fase no minima
if isempty(Z) | max(abs(Z))<(1-TOLER)
    Gx=1/Go;
else
    %ordenem els zeros en ordre creixent de abs
    Z=sort(Z);
    %zeros de fase minima
    i=1;
    num_z_fm=0;
    while 1-abs(Z(i))>TOLER
        z_fm(i)=Z(i);
        num_z_fm=num_z_fm+1;
        i=i+1;
    end
    if num_z_fm>0
        z_fm=z_fm';
    end
    %zeros de fase no minima
    while i<=size(Z,1)
        z_fnom(i-num_z_fm)=Z(i);
        i=i+1;
    end
    z_fnom=z_fnom';
    num_z_fnom=size(Z,1)-num_z_fm;

    %CALCUL DE Gx
    %els pols de Go passen a ser zeros de Gx
    [nGo,dGo]=tfdata(Go,'v');
    D=tf(dGo,1,Ts);

    %els zeros de fase minima z_fm de Go passen a ser pols de Gx
    if num_z_fm>0
        Np=zpk(z_fm,[],1,Ts);
    else
        Np=1;
    end

    %repetim els zeros de fase no minima z_fnom
    Nm1=zpk(z_fnom,[],1,Ts);
    [nNm1,dNm1]=tfdata(Nm1,'v');
    nNm2=fliplr(nNm1);
    dNm2=zeros(1,num_z_fnom+1);
    dNm2(1)=1;
    Nm2=tf(nNm2,dNm2,Ts);

    %calcul de bmax
    Nm=Nm1*Nm2;
    %com que, en general, Nm es no causal, afegim retards z^-1 que no
    %influeixen en el guany de Nm
    [nNm,dNm]=tfdata(Nm,'v');
    dNm=zeros(1,size(dNm,2));
    dNm(1)=1;
    Nm=tf(nNm,dNm,Ts);
    b=norm(Nm,inf);

    %incorporem el guany de Go
    [z,p,gain]=zpkdata(Go,'v');

    %Gx
    Gx=D*Nm2/(Np*gain*b);
end
```

### \*\*\* espectrepot\_resposta.m \*\*\*

```
function []=espectrepot_resposta(time,signal,initial_time,final_time);

%Aquesta funcio dibuixa l'espectre de potencia del senyal signal a partir
%de les dades compreses entre initial_time i final_time.

%es defineix una tolerancia per problemes de quantificacio
TOLER=1e-10;

%es remostreja signal amb un vector de temps amb increments constants
Ts=0.005;
time_spline=0:Ts:final_time;
signal_spline=spline(time,signal,time_spline);

%s'agafa l'interval d'interes
M=find( abs(time_spline-(initial_time+Ts))<TOLER );
for i=1:((final_time-initial_time)/Ts)
    time_sampled(i)=time_spline(M-1+i);
    signal_sampled(i)=signal_spline(M-1+i);
end
time_sampled=time_sampled';
signal_sampled=signal_sampled';

%s'elimina la component continua
signal_sampled=detrend(signal_sampled,'constant');

%es calcula l'espectre de potencia del senyal
N=size(signal_sampled,1);
SIGNAL=fft(signal_sampled,N);
Psignal=SIGNAL.*conj(SIGNAL)/(N*N);
%
freqs=(1/Ts)*(0:N/2-1)/N;

%es dibuixa l'espectre en eixos lineals
plot(freqs,Psignal(1:N/2),'b');
grid;
```

## **Annex B**

### **ÍNDEX CD**

En aquest annex es detalla l'índex del CD que s'adjunta a aquest treball, amb la relació de les carpetes i els arxius per tal de facilitar-ne la localització.

```

C_REPE\
  matlab\
    o identificació\
      ▪ corba_estàtica\
        • plot_c_estat.m
        • sys_gain.m
        • c_estat_down.out
        • c_estat_up.out

      ▪ prbs\
        • espectre_prbs.m
        • prbs_adapt.m
        • wsp_prbs_data_input.mat
        • model_prbs.sid
        • prbs_iden.out
        • prbs_valid.out

      ▪ resposta_temporal\
        • plot_resposta.m
        • sys_tau.m
        • resposta.out

    o model_dsa\
      ▪ control_velocitat\
        • espectrepot_resposta.m
        • parell.m
        • module.m
        • tf_repe.m
        • gx.m
        • wsp_planta.mat
        • wsp_filtres.mat
        • control.mdl

      ▪ planta\
        • sim_dsa.mdl

    o model_físic\
      ▪ planta\
        • espectrepot_resposta.m
        • parell.m
        • module.m
        • wsp_planta.mat
        • wsp_planta_pert.mat
        • planta.mdl
        • planta_pert.mdl

    o model_toolbox\
      ▪ control_velocitat\
        • espectrepot_resposta.m
        • parell.m
        • module.m
        • tf_repe.m
        • gx.m
        • wsp_planta.mat
        • wsp_filtres.mat
        • control.mdl

      ▪ planta\
        • sim_toolbox.mdl

```

memòria.pdf